I P K S

Prozess - Software - Entwicklungs GmbH

M2000 + SICOMP M/R Emulation

für Windows

Benutzerhandbuch

Copyright 1998/2007
© IPKS GmbH

Alle Informationen in dieser Unterlage sind Eigentum von IPKS.
Weitergabe, Verwertung sowie Vervielfältigung - auch auszugsweise - ist nur nach ausdrücklicher Genehmigung durch IPKS gestattet. Zuwiderhandlungen verpflichten zu Schadensersatz.

Einschränkung der Gewährleistung

Es wird keine Garantie für die Richtigkeit dieses Handbuches übernommen, da sich Fehler, trotz aller Bemühungen, nie vollständig vermeiden lassen.

Nutzungsrechte

Die Vertragsbedingungen für die Benutzung von IPKS-Software durch den Lizenznehmer werden vor der eigentlichen Softwareinstallation zum Lesen und Quittieren vorgelegt.

Herausgeber

IPKS Prozess-Software-Entwicklungs GmbH

Am Weichselgarten 36
91058 Erlangen
Tel.: 09131/69588-0
Fax.: 09131/69588-25
E-mail: support@ipks.de

Stand

19.02.2007

Programmversion

Das Handbuch beschreibt M2000 (ab Version 4) und SICOMP M/R Emulation

Warenzeichen

Wir weisen darauf hin, dass die in dieser Beschreibung verwendeten Soft- und Hardwarebezeichnungen und Markennamen im allgemeinen warenzeichen-, marken- oder patentrechtlichem Schutz unterliegen.

INHALT

1 Übersicht 4

2 Funktionsbeschreibung 7

2.1 Softwarestruktur 7
2.2 Leistungsumfang 8
2.2.1 CPU-Emulation 8
2.2.2 Geräteemulationen 8

3 Inbetriebnahme 16

3.1 Lieferumfang 16
3.2 Systemvoraussetzungen 19
3.3 Installation 20
3.3.1 Übersicht 20
3.3.2 M2000 (incl. Emulator) installieren 22
3.3.2.1 Anwender-System installieren 23
3.3.2.2 Änderung von SystemOptionen 23
3.3.3 Konfigurationsdatei (mpar.sys/rpar.sys) 24
3.4 Bedienung des Emulators mit M2000 29

4 Konfigurationsdatei für SIC-M 30

4.1 Allgemeine Einträge 30
4.2 Einträge für Gerätedefinitionen 32
4.2.1 Sichtgeräte 32
4.2.2 Speichermedien 38
4.2.3 Drucker 41
4.2.4 Kopplungen 49
4.2.5 Sonstige Geräte 58
4.2.5.1 ZeitSynchronisation 58
4.2.5.2 Zeitgeber 59
4.2.5.3 Profibus & Pseudo Geräte 60
4.2.5.4 Promea MX & BDE Terminal (ES 1) 65

5 Konfigurationsdatei für SIC-R 68

5.1 Allgemeine Einträge 68
5.1.1 Sichtgeräte 70
5.1.2 Speichermedien 74
5.1.3 Drucker 77
5.1.4 Kopplungen 84
5.1.5 Sonstige Geräte 90

6 Details zu den Geräteemulationen 97

6.1 Geräteemulation DS074 97
6.1.1 Bildschirmaufbau 97
6.1.2 Tastenbelegung 98
6.2 Rechnerkopplung DU05 100
6.2.1 Emulation DU05 über DF42 100
6.2.1.1 Voraussetzung 100
6.2.1.2 Parametrierung 100
6.2.1.3 Protokollparameter für MSV2 101
6.2.2 Umlenkung der DU05 Kommunikation auf TCP/IP 101
6.3 Rechnerkopplung DU06 102
6.3.1 Emulation DU06 über DF32/42 102
6.3.1.1 Voraussetzung 102
6.3.1.2 Parametrierung 103
6.3.1.3 Protokollparameter für HDLC-UNC 104

© IPKS GmbH 2007

6.3.1.4 Protokollparameter für HDLC-BAC ... 104
6.4 Offene DU0x-Kommunikations-Schnittstelle ... 105
6.4.1 Überblick ... 105
6.4.2 Umlenkung der DU04 Kommunikation auf TCP/IP ... 105
6.4.3 Umlenkung der DU05 Kommunikation auf TCP/IP ... 105
6.4.4 Umlenkung der DU06 Kommunikation auf TCP/IP ... 107
6.5 Kommunikationssteuerung CP1400 ... 108
6.5.1 Installation ... 108
6.6 Kommunikationssteuerung CS275 ... 109
6.6.1 Installation der Baugruppe ... 109
6.6.2 Software Installation ... 109
6.7 Rechnerkopplung DUST3965R ... 110
6.7.1 Emulation DUST3965R über DF42 ... 110
6.7.1.1 Voraussetzung ... 110
6.7.1.2 Parametrierung ... 110
6.7.1.3 Protokollparameter für MSV2 ... 111
6.8 Rechnerkopplung DUST3961, GatewayFunktion ... 112
6.9 Rechnerkopplung DUST 3966 ... 113
6.9.1 Emulation DUST 3966 über DF32/42 ... 113
6.9.1.1 Voraussetzung ... 113
6.9.1.2 Parametrierung ... 113
6.9.1.3 Protokollparameter für HDLC-UNC ... 114
6.9.1.4 Protokollparameter für HDLC-BAC ... 115
6.9.2 Umlenkung der DUST 3966 Kommunikation auf TCP/IP ... 115
6.9.3 Umlenkung der DUST3964R Kommunikation auf TCP/IP ... 115
6.10 Emulation PE3600 über ProfiBus oder SFT direkt ... 116
6.10.1 M2000 ... 116
6.10.2 Konfiguration *rpar.sys* ... 116
6.10.3 Beispiel ... 119
6.11 PE3600 – Siemens SFT Baugruppe ... 119
6.11.1 Beispiel ... 119
6.12 Schnittstellenkonzentrator ... 120
6.12.1 Allgemeines ... 120
6.12.2 *mpar.sys* Parametrierung ... 120
6.12.3 Stellvertreterprozesse auf dem entfernten Rechner ... 121
6.13 OP31 Emulation ... 122
6.13.1 Inbetriebnahmehinweise zum Laufwerk SONY RMO-S580 ... 122
6.13.1.1 Funktionsschalter an der Rückwand ... 122
6.13.1.2 Inbetriebnahme ... 122
6.13.2 Wichtige Hinweise ... 123
6.13.3 Datenträger ... 123
6.14 UNKLAR Erkennung bei seriellen Schnittstellen ... 124
6.14.1 TTY / V.24 Schnittstellen ... 124
6.14.2 V.24-Anschluss ... 124
6.15 Kompatibilitätseinschränkungen ... 125
6.15.1 zum M-Rechner ... 125
6.15.2 zum R-Rechner ... 127
6.16 Kommunikationskanal SCU – SICOMP M/R Programme ... 128

7 Zusatzprogramme ... 129

7.1 Programm PCCOPY ... 129
7.2 Programm MCSAVE.EXE ... 130
7.3 Programm WAIT4.EXE ... 131
7.4 Programm SOLDIS ... 132
7.5 SicView, M2000 Festplatten Explorer ... 133

 © IPKS GmbH 2007

8 Emulation PE F7 134
8.1 Hardwarestruktur 134
8.1.1 SICOMP M 134
8.1.2 M2000 135
8.2 Softwarestruktur 136
8.2.1 Zugriff auf die Prozessperipherie 137
8.2.2 Alarmbearbeitung 137
8.3 Funktionsbeschreibung 138
8.3.1 Geräteemulation 138
8.3.1.1 Elementaraufrufe 138
8.3.1.2 Gerätefunktion 138
8.3.1.3 Adressierung 138
8.4 ProfiBus DLL 139
8.4.1 Allgemeines 139
8.4.2 Installation (ProfiBus) 139
8.5 Parametrierung 140
8.5.1 Konfigurationsdatei *mpar.sys* 140
8.5.2 Beispiel (ProfiBus) 141
9 Anhang 142
9.1 Allgemeine Hinweise 142
9.2 Meldungen 143
9.2.1 Start- / Ende- / Fehler - Meldungen. 143
9.2.2 Tracemeldungen. 143
9.2.2.1 TrcView-Fenster 143
9.2.2.2 TrcView Leistungen 143
9.3 Muster der Parameterdatei *mpar.sys* 144
9.4 Muster der Parameterdatei *rpar.sys* 150
9.4.1 Muster des StartBatch *emu_strt.bat* 153
9.5 M2000 Fehler-MeldeFormular 154
9.6 Fehler-Analyse durch IPKS GmbH 155
10 Index 156

1 Übersicht

Siemens Minicomputer SICOMP M/R sind Rechner, die sich weltweit im industriellen und kommerziellen Einsatz durch ihre Flexibilität, hohe Verfügbarkeit und - nicht zuletzt - leistungsfähige Systemsoftware bewährt haben. Für diese Rechner wurde in der Vergangenheit, mit hohen finanziellen Aufwendungen, umfangreiche Anwendungs-Software entwickelt, die ihre Stabilität und Brauchbarkeit bereits bewiesen hat und in vielen Fällen auch für die weitere Zukunft durchaus ausreichend wäre.
Durch die gewaltigen Fortschritte in der Hardware-Entwicklung ist jedoch die PC-Technik heute in alle Computer-Anwendungsbereiche vorgedrungen und hat sich zum Hardware-Standard entwickelt. Dies gilt auch für die industriellen Anwendungen.

Die SICOMP M/R - Anwendungen können an diesem Fortschritt in der Hardware-Entwicklung nicht partizipieren. Sie als Betreiber der SICOMP M/R Anlage möchten aber Vorteile aus dieser Entwicklung ziehen, wie z.B.:

- Nutzung kostengünstiger PC-Hardware
- Nutzung der Fortschritte in der Hardware-Entwicklung
- Reduzierung der Wartungskosten
- Integration Ihrer Anwendung in die PC- bzw. Windows-Welt

M2000, der Emulator für SICOMP M/R unter dem Betriebssystem Windows[1] bietet Ihnen die Lösung Ihres Problems. Der Emulator setzt den Hardware-Befehlssatz und die Ebenenstruktur eines Siemens Minicomputers SICOMP M/R auf den Befehlssatz der Intel x86 Prozessoren (oder kompatiblen) um. Damit ist es möglich, ein auf einem Siemens SICOMP M/R System generiertes, lauffähiges System auch auf einem PC ablaufen zu lassen.

[1] Windows = Windows2000, Windows2003Server, Windows XP

© IPKS GmbH 2007

Bild 1 zeigt in einer Gegenüberstellung ein Siemens SICOMP M/R Ursprungssystem und das gleiche System mit M2000 auf einem PC ablaufend. M2000 übernimmt die Anpassung des Siemens SICOMP M/R Hardware-Befehlssatzes an den Befehlssatz des Intel Prozessors und die Anbindung an die Peripherie des PC.

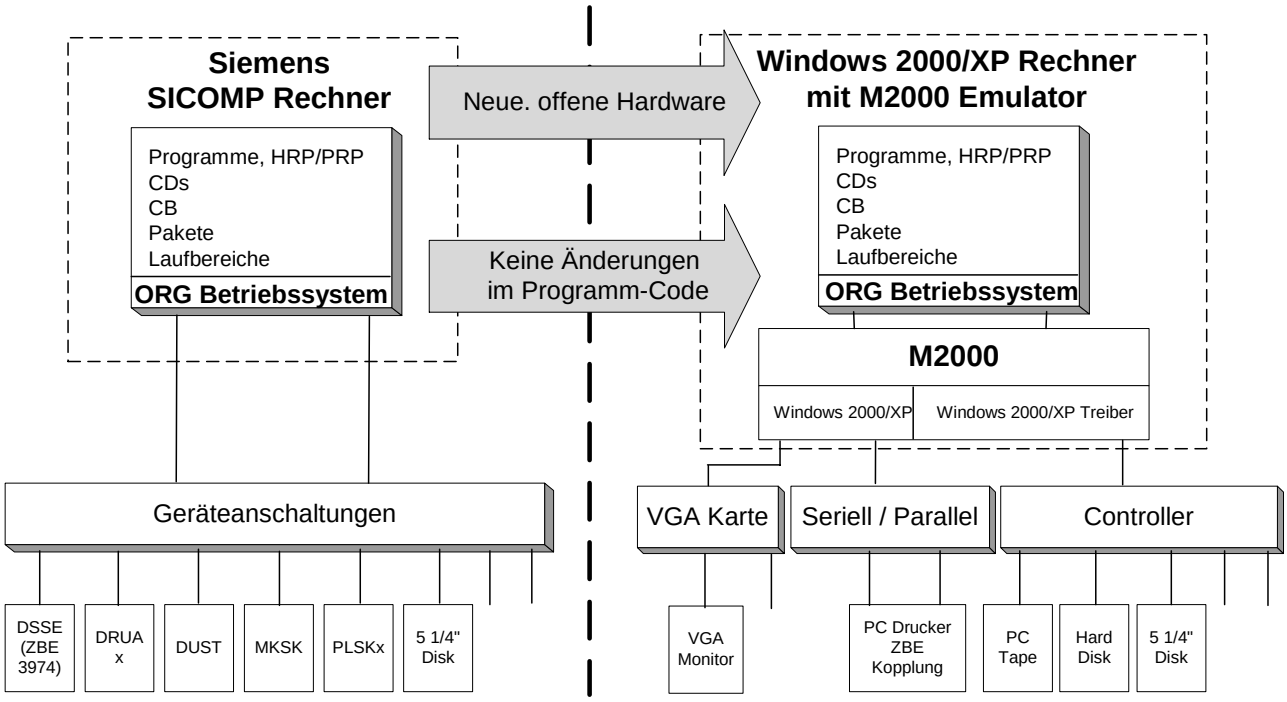


Bild 1: Gegenüberstellung SICOMP - M2000

Ein-/Ausgabe Aufrufe des Betriebssystems ORG-M/ORG-R und von Siemens SICOMP Programmen werden von den M2000-E/A Treiberprozessen bearbeitet. E/A-Aufrufe für Plattenspeicher (PLSK) werden auf Windows-Dateien umgelenkt und dabei die PLSK nachgebildet. E/A-Aufrufe zu Geräten wie Drucker (DRUA), Sichtgerät (DSSK) usw. werden von Emulationsprozessen bearbeitet.

© IPKS GmbH 2007

Performance

Prinzipiell bedeutet jede Emulation einen Performance-Verlust; doch die Leistungsfähigkeit der heute angebotenen Prozessoren und Peripheriekomponenten ist wesentlich größer als die der SICOMP-Minicomputer, so dass die Emulation bei entsprechender PC-Basis zu einer Leistungsverbesserung führt.
Wird ein großes SICOMP System (Anzahl Geräte) abgelöst, dann sollten die PC, auf denen die Emulation läuft, eine entsprechende Performance bieten.
Im Einzelnen heißt das:

- 2 Prozessoren (Dual Core) mit x GHz (akt. Stand der Technik)
- 1024 MByte Speicher
- schnelle Festplatten, ausreichend FP Speichervolumen

In laufenden Anlagen hat sich gezeigt, dass man mit aktuellen Industrie-PC eine Performance erreichen kann, die der mehrfachen Leistung eines SICOMP M80 entspricht.

Anwendungen, die einen hohen Anteil von Plattenzugriffen durchführen, erreichen in der Regel nochmals eine mehrfache Leistungssteigerung der oben angegebenen Leistungswerte, da die heute verwendeten Platten (RAID Systeme) wesentlich schneller sind und Windows darüber hinaus noch eine sehr effiziente Zugriffsoptimierung besitzt.

 © IPKS GmbH 2007

2 Funktionsbeschreibung

2.1 Softwarestruktur

M2000, der Emulator für SICOMP M/R besteht aus einem Emulatorkern-Prozess und verschiedenen Peripherie-Prozessen. Diese Prozesse werden unter Windows installiert und laufen, entsprechend ihrer Priorität, konkurrierend zu anderen Windows-Prozessen.

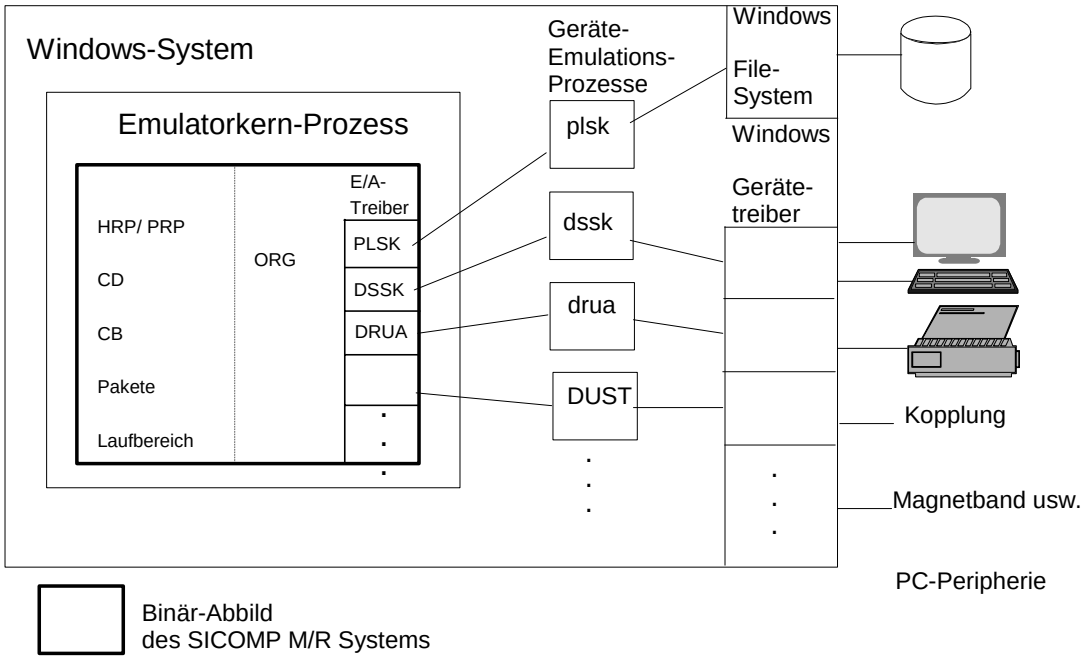


Bild 2: Softwarestruktur

Der Emulatorkern-Prozess bearbeitet die SICOMP M/R Befehle, vergleichbar mit dem Microcode eines SICOMP M/R Prozessors, und bildet die Hardware-Prioritätsebenen und die Unterbrechungssteuerung eines SICOMP M/R Rechners nach. Dadurch erfolgt die Prozessverwaltung wie auf einem originalen SICOMP M/R. Die Kommunikation zwischen 'ZE' (Emulatorkern-Prozess) und Peripherie (Peripherie-Prozessen) wird über die gleichen Prozessblöcke des SICOMP Systems abgewickelt.
Die Geräteemulations-Prozesse bilden die SICOMP Geräte auf die Geräte des Windows-Systems ab - z. B. wird ein Teilgerät eines Plattenspeichers in einer Datei im Windows-Dateisystem abgebildet. Durch diese logische Abbildung ist die Art des physikalischen Geräts nicht relevant.

Die Emulation bildet konsequent die Hardware-Eigenschaften einer SICOMP M/R Zentraleinheit nach. Deshalb ist es unerheblich, welche Software oder welches Betriebssystem das zu emulierende Anwendersystem enthält. Es muss lediglich ein unter SICOMP M/R ablauffähiger Code sein und die verwendete Peripherie muss durch einen Peripherie-Prozess von M2000 unterstützt werden.
(siehe Kapitel: Leistungsumfang)

Der Emulatorkern-Prozess emuliert auch die virtuelle Konsole.

© IPKS GmbH 2007

2.2 Leistungsumfang

2.2.1 CPU-Emulation

SIC-M/R

M2000 emuliert den Befehlssatz der SICOMP M/R Zentraleinheiten (ZE) und unterstützt dabei die Leistungen der einzelnen ZE. Durch die Wahl der zu emulierenden ZE (ZE01...ZE03, R10, R10V, R20, R30) wird genau der Befehlssatz und der Leistungsumfang dieser ZE zur Verfügung gestellt.
Die Gleitpunktarithmetik wird ebenfalls ZE-spezifisch emuliert. Dabei erfolgt keine Nutzung der Floating-Point Arithmetik des Intel-Prozessors, sondern es wird die Gleitpunktarithmetik des SICOMP M/R-Rechners nachgebildet. Nur dadurch kann sichergestellt werden, dass die Genauigkeit des jeweiligen ZE-Typs exakt eingehalten wird.

2.2.2 Geräteemulationen

Die Schnittstelle zwischen CPU-Emulation und Geräteemulation ist der EA-Auftrag, der von den Geräteemulationen interpretiert und bearbeitet wird. Dabei emulieren die Geräteemulationen einen gewissen Teil der Firmware der Geräteanschaltungen des SICOMP M/R.

SIC-M

Die EA-Auftragsbearbeitung erfolgt gemäß der SICOMP M Kommunikations-Schnittstelle (KOSS).
Bei Bearbeitung des SICOMP M-Befehls EAS aktiviert die CPU-Emulation die Geräteemulation.

Sichtgeräte

Die Sichtgeräte werden grundsätzlich über serielle Schnittstellen angeschlossen. Dazu dient eine unter Windows installierte serielle Schnittstelle COM1: bis COM9: oder \\.\COM10..999.
Folgende Sichtgeräte bzw. deren Emulationen werden über serielle Schnittstelle unterstützt:

- ZBE3974R, ZBE3974MT
- DS075, DS075F
- DS075-DISIT
- DS078, DS081
- VDU2000

Für DS075 Geräte kann sich die serielle Schnittstelle auch auf einem entfernten, über LAN verbundenen Windows PC befinden.
(siehe Kapitel: Schnittstellenkonzentrator).

Sichtgeräte-Emulationen

IPKS bietet im Rahmen von M2000 eine Sichtgeräteemulation an,
die folgende Leistungen bietet:

- Alpha-Terminal Emulation TE2000 AX
  TE2000 AX emuliert die Sichtgeräte ZBE3974R bzw. DS075
- Alpha-Terminal Emulation TE2000 FG (mit erweitertem Funktionsumfang)
  TE2000 FG emuliert die Sichtgeräte DS075 FG
- DISIT-Terminal Emulation TE2000 DX
  TE2000 DX emuliert die Sichtgeräte ZBE3974MT bzw. DS075 DISIT
- Terminal Emulation TE2078
  TE2078 emuliert das Sichtgerät DS078

Die Sichtgeräte Emulationen können sowohl im Fenstermodus als auch im Vollbildmodus betrieben werden.

Sie können in Verbindung mit M2000 wie folgt genutzt werden:

- lokal auf dem gleichen System wie M2000 unter Windows
- auf einem separaten PC, über seriellen Anschluss
- auf einem separaten PC, über LAN-Verbindung

Hinweis

Ausführliche Beschreibungen in den Benutzerhandbüchern:
TE2000/TE2078 Terminal-Emulation

© IPKS GmbH 2007

Konsolen-Emulation

Die Windows-Konsole, d.h. der Bildschirm und die Tastatur des PC, kann als virtuelle Konsole, Standardmelde- bzw. Standardbediengerät verwendet werden.
M2000 enthält dafür eine Geräteemulation DS074. Diese kann sowohl im Window-Modus als auch im Vollbild-Modus betrieben werden. Im Window-Modus kann diese Terminal-Emulation auch mehrfach aktiviert werden.
Die Konsolen-Emulation beinhaltet eine Untermenge der Funktionen der Alpha-Terminal Emulation TE2000 AX. Wenn von der Konsole aus anspruchsvollere SICOMP Anwendungssysteme bedient werden sollen, die beispielsweise parametrierbare/ladbare Funktionstasten verwenden, so muss hier die Terminal Emulation TE2000 AX bzw. TE2000 FG eingesetzt werden.

Festplatten

Die PLSK-Geräteemulation bildet ein Teilgerät einer SICOMP M/R-Festplatte auf eine Windows-Datei ab. Diese Datei repräsentiert eine SICOMP M/R-Platte.
Nach dem Kopieren der Platten per Magnetbandkassette oder optischer Platte in die Datei, enthält diese die gleichen Daten, einschließlich der Datenträgerbuchführung, wie die M/R-Platte.
Die Schnittstelle ist der EA-Auftrag. Damit ist gewährleistet, dass die Verwaltung der Peripheriespeicher nicht emuliert werden muss, sondern als Bestandteil des portierten SICOMP M/R-Systems unter der CPU-Emulation abläuft.

Der Bediener erkennt in der Regel keinen Unterschied zu einem Original-Laufwerk:

- Der Datenzugriff auf die Platten erfolgt in der gleichen Reihenfolge wie bei SICOMP M/R.
- Die Verwendung des Parameters **mode=share** in der Konfigurations-Datei *mpar.sys*/*rpar.sys* ermöglicht den Zugriff auf die Platten aus beiden Systemumgebungen (SICOMP und Windows). Fehlt der Parameter, ist der Belegmodus *exclusiv* eingestellt (default).
  Der Modus =*share* muss verwendet werden, wenn mit dem Windows Programm *sicview.exe* (siehe Kap. Zusatzprogramme) auf Platten eines emulierten SICOMP Systems zugegriffen werden soll.
  Der Modus kann weiterhin verwendet werden, um Platten eines emulierten SICOMP Systems unter Windows zu sichern, ohne den Emulator zu beenden. Hierbei muss aber darauf geachtet werden, dass beispielsweise DVS Dateien geschlossen sind bzw. das Anwendersystem sinnvollerweise nicht gestartet ist.
- Die Verwaltung des Datenträgerinhalts erfolgt weiterhin über die SICOMP M/R Dateiverwaltung bzw. DVS.
- DVS-Datenträger werden ebenfalls bearbeitet.
- Die PLSK-Emulation benutzt das Windows-Dateisystem. Durch intelligentes Plattencaching und schnellere Platten, Controller bzw. SCSI-Bus sind sowohl die Zugriffs-, als auch die Transfergeschwindigkeiten wesentlich höher als auf dem Originalsystem. Bei der Verwendung von RAID-Systemen wird zusätzlich noch eine hohe Datensicherheit erreicht.

Der direkte Anschluss von SICOMP-M/R-Laufwerken an den PC mit Emulator ist nicht möglich.

Drucker

Anschließbar sind alle Drucker mit Centronics-Schnittstelle, entsprechend den Erfordernissen der Anwendung. Ebenso sind betreibbar:

- DR202-Drucker
- 3915 Tally
- 3916 Trommeldrucker
- 3918 UD3
- 3919 PT80i

Drucker DR202 sind auch über die serielle Schnittstelle betreibbar.
Die serielle Schnittstelle befindet sich am lokalen M2000 PC oder auf einem entfernten, über LAN verbundenen Windows PC.
(siehe Kapitel: Schnittstellenkonzentrator).

© IPKS GmbH 2007

Hinweis

Es sollten nur Drucker verwendet werden, die standardmäßig im ASCII-Modus mit eigenem Zeichensatz betrieben werden können.

Hardcopy-Ausgaben können mit M2000 auf einen EPSON-kompatiblen Drucker am PC ausgegeben werden. Dazu muss in FACO30 auch EPSON eingestellt sein.

Druckersteuerzeichen (z.B. für Schriftartenumschaltung) werden von M2000 unverändert an den Drucker weitergegeben. Gegebenenfalls muss hier eine Anpassung von Anwendungsprogrammen erfolgen, wenn in Verbindung mit M2000 nicht der gleiche oder ein kompatibler Drucker zum Ursprungssystem verwendet werden kann.

Zeitgeber

Die ZIG-Geräteemulation bildet EA-Aufträge an den Zeitgeber auf entsprechende Windows-Aufrufe ab. Durch die verwendete Standard-Hardware und den Einsatz von Windows sind bei zeitmodifizierten Aufrufen Einschränkungen hinsichtlich der Granularität der Uhrzeit und der kleinsten möglichen zyklischen Weckzeit gegeben. Die kleinste Zeiteinheit beträgt 10ms, bei Einsatz von ZIG3 1ms Auflösung.

Magnetband-Kassetten-Laufwerk

M2000 betreibt das Magnetband-Kassetten-Laufwerk des PC als MK82 Emulation. Es können 150MB-Kassetten gelesen und geschrieben werden. Der Daten austausch mit einem SICOMP System ist damit möglich.

Für die Verwendung der angegebenen kompatiblen Magnetband-Kassetten-Laufwerke gelten folgende Aussagen:

- physikalisches Lesen und Schreiben ist möglich
- logisches Lesen und Schreiben ist möglich
- Lesen und Schreiben von DVS-Volumes ist möglich

Anmerkung

Mit MK80/81 Laufwerken erzeugte Magnetband-Kassetten können von der M2000 MK82-Emulation gelesen werden. Die am PC betriebenen Magnetbandkassetten-Laufwerke sind aber nicht in der Lage, das MK81-Format zu schreiben.

Floppy-Disk

M2000 betreibt das 3,5"(1.44MB) oder 5,25"(1,2MB) Diskettenlaufwerk als FD044 mit der SICOMP M/R Initialisierung. Ein Datenaustausch mit einem SICOMP M/R System ist jedoch nicht möglich.

Rechnerkopplungen

DU02

Die Gateway-Funktionalität zur DU02 (Lichtwellenleiter-Kopplung) ist implementiert. Damit lassen sich zwei Emulator-Systeme verbinden.

DU03

M2000 ermöglicht die Emulation einer Rechnerkopplung DU03 über einen speziellen Treiber(NDIS/NDIS2000). Auf dem Emulator-PC muss ein Paket- oder NDIS-Treiber installiert werden, der von IPKS derzeit für WindowsNT und Windows2000 geliefert werden kann. Die DU03-Emulation bildet die Original-Funktionalität -Datentransfer- vollständig ab. Nicht abgebildet sind Test- und Wartungsaufträge, die aber aus Kompatibilitätsgründen anzeigenfrei durchlaufen und abgeschlossen werden.
Die DU03 wird ohne spezielle Hardware emuliert, es wird lediglich eine handelsübliche Netzwerkkarte benötigt. Daraus resultiert, dass eine emulierte DU03 auch mit 100 Mbit/s arbeiten kann.
Mit Hilfe des Treibers kann die DU03-Emulation über die zugehörige Netzwerkkarte jedes empfangene Datenpaket lesen, auf Relevanz prüfen und verarbeiten, sowie Datenpakete versenden.

 © IPKS GmbH 2007

DU04

M2000 ermöglicht die Emulation einer Rechnerkopplung DU04 über eine serielle Schnittstelle. Dabei werden drei Möglichkeiten angeboten:

- Emulation über eine beliebige serielle Schnittstelle (COMxx)
- Emulation über die Schnittstellenbaugruppe DF32/42
- Emulation über TCP/IP im LAN oder über RAS.

Die DU04-Geräteemulation ermöglicht per Parametrierung auch eine Kommunikation über TCP/IP. Diese bietet sich z.B. dann an, wenn zwei M2000 Systeme miteinander kommunizieren sollen oder ein Datenaustausch zu Fremdsystemen (UNIX) realisiert werden soll.

In diesem Fall wird in AMBOSS/BS-M eine DU04 generiert. Die Geräteemulation setzt diese Kommunikation dann auf die Windows-Socket-Schnittstelle(Winsocket) um. Anstelle der COMxx Schnittstelle oder der DF32/42 wird ein LAN/RAS Netzwerk-Adapter gemäß Windows Hardware Kompatibilitätsliste verwendet.

DU05

Emulation einer Rechnerkopplungsschnittstelle DU05. Die DU05 Emulation bearbeitet das Übertragungsprotokoll MSV2.

Damit sind beispielsweise folgende Kopplungen realisierbar:

- M2000 - M2000 über MSV2 und SINEC
- M2000 über MSV2 zu Fremdsystemen
- M2000 über TCP/IP im LAN oder über RAS.

Für die Emulation der DU05 ist eine Datenübertragungsbaugruppe DF42 mit dem entsprechenden Protokolltreiber erforderlich.

Die DU05-Geräteemulation ermöglicht per Parametrierung auch eine Kommunikation über TCP/IP. Diese bietet sich z.B. dann an, wenn zwei M2000 Systeme miteinander kommunizieren sollen. In diesem Fall wird in AMBOSS/BS-M eine DU05 generiert. Die Geräteemulation setzt diese Kommunikation dann auf die Windows-Socket-Schnittstelle (Winsocket) um. Anstelle der DF42 wird ein LAN/RAS Netzwerk-Adapter gemäß Windows Hardware Kompatibilitätsliste verwendet.

DU06

Emulation einer Rechnerkopplungsschnittstelle DU06. Die DU06 Emulation bearbeitet die Übertragungsprotokolle HDLC U/P, HDLC U/S und HDLC B.

Damit sind beispielsweise folgende Kopplungen realisierbar:

- M2000 - M2000 über HDLC B und HDLC U und SINEC
- M2000 - IBM über SDLC und SINEC SNSNA
- M2000 - X25 Netz über HDLC B und SINEC SNPV
- M2000 über TCP/IP im LAN oder über RAS.

Nicht implementiert sind die WIKO-Funktionen (Remote-Laden...).

Für die Emulation der DU06 ist eine Datenübertragungsbaugruppe DF32 oder DF42 mit dem entsprechenden Protkolltreiber erforderlich.

Die DU06-Geräteemulation ermöglicht per Parametrierung auch eine Kommunikation über TCP/IP. Diese bietet sich z.B. dann an, wenn zwei M2000 Systeme miteinander kommunizieren sollen. In diesem Fall wird in AMBOSS/BS-M eine DU06 B generiert. Die Geräteemulation setzt diese Kommunikation dann auf die Windows-Socket-Schnittstelle (Winsocket) um. Anstelle der DF32/42 wird ein LAN/RAS Netzwerk-Adapter gemäß Windows Hardware Kompatibilitätsliste verwendet.

© IPKS GmbH 2007

PARIF Kopplung (Ersatz)

3961 Gateway-Funktion zu PARIF.(**PAR**allel**I**nter**F**ace)
Mit dieser Kopplung werden eine Sende- und eine Empfangsverbindung zu einem Gateway-PC definiert, der über eine PARIF-Baugruppe mit einem Original R-Rechner kommuniziert. Im emulierten ORG wird diese Kopplung als DUST 3961 generiert und als KNWE/KNWA betrieben.

PE3600

Die Prozesseinheit wird über Profibus Peripherie aus dem SIMATIC S7 Spektrum emuliert. Damit kann die PE3600 durch diese moderne und kostengünstige standardisierte Technik abgelöst werden, ohne dass Änderungen in der SICOMP R Software erforderlich sind.

Der Anschluss des Profibus erfolgt über einen ProfiBus Kommunikationsprozessor und der zugehörigen Software aus dem SIMATIC NET Spektrum.

Die Profibus Peripheriebaugruppen werden als dezentrale Peripherie im System ET200 betrieben.

Derzeit werden folgende Prozesssignalformer unterstützt:

- Digitaleingabe 3611, 3612, 3613, 3615
- Digitalausgabe 3621, 3622, 3625
- Analogeingabe 3631
- Analogausgabe 3652
- Prüfmelder 3668 (nicht funktional, wird simuliert)

PE F7

Die Prozesseinheit PE F7 eines SICOMP M wird über den Profibus emuliert.
M2000 enthält eine **PE F7** Geräteemulation für die E/A Zugriffe auf die Prozessperipherie. Diese Gerätemulation bearbeitet direkt die Prozessaufrufe, die nicht auf ein im ORG-M generiertes Prozessgerät führen. Prozessaufrufe, die das Prozessgerät ALEM verwenden, werden von einer M2000 Gerätefunktion **ALEM** bearbeitet. Siehe auch: Kapitel 8, Emulation PE F7

Derzeit werden folgende Prozesssignalformer unterstützt:

- Digitaleingabe 430
- Digitalausgabe 451
- Digitalein-/ausgabe DEDA 482
- Analogeingabe 460
- Analogausgabe 470
- Zählerbaugruppe IP 242 A (eingeschränkte Funktionalität)

 © IPKS GmbH 2007

PROMEA1

Geräte, die am Originalsystem über die PROMEA1 Baugruppe angeschlossen sind, werden am PC an die COM Schnittstellen angeschlossen und im Emulator über die Geräteprozesse für Drucker, Sichtgerät und DUST3964R betrieben.

PROMEA MX

M2000 ermöglicht die Emulation einer parametrierbaren PROMEA.
Der Betrieb erfolgt über eine serielle Standard-Schnittstelle (COMxx). Die serielle Schnittstelle befindet sich am lokalen M2000 PC oder auf einem entfernten, über LAN verbundenen Windows PC (siehe Kapitel: Schnittstellenkonzentrator).

Rechnerkopplung

DUST3961

Mit der Gateway-Funktionalität zur DUST 3961 lassen sich zwei EmulatorSysteme verbinden.

DUST3962

Mit der Gateway-Funktionalität zur DUST 3962 (Lichtwellenleiter-Kopplung) lassen sich zwei EmulatorSysteme verbinden.

DUST3964R

M2000 ermöglicht die Emulation einer Rechnerkopplung DUST3964R über eine serielle Schnittstelle. Dabei werden zwei Möglichkeiten angeboten:

- Emulation über eine beliebige serielle Schnittstelle (COMxx)
- Emulation über TCP/IP im LAN oder über RAS.

Die DUST3964R-Geräteemulation ermöglicht per Parametrierung auch eine Kommunikation über TCP/IP. Diese bietet sich z.B. dann an, wenn zwei M2000 Systeme miteinander kommunizieren sollen.
In diesem Fall wird im ORG eine DUST3964R generiert. Die Geräteemulation setzt diese Kommunikation dann auf die Windows-Socket-Schnittstelle (Winsocket) um. Anstelle der COMxx Schnittstelle wird ein LAN/RAS Netzwerk-Adapter gemäß Windows Hardware Kompatibilitätsliste verwendet.

DUST3965R

Emulation einer Rechnerkopplungsschnittstelle DUST3965R. Die DUST3965R Emulation bearbeitet das Übertragungsprotokoll MSV2.

Damit sind beispielsweise folgende Kopplungen realisierbar:

- M2000 - M2000 über MSV2 und SINEC
- M2000 über MSV2 zu Fremdsystemen
- M2000 über TCP/IP im LAN oder über RAS

Für die Emulation der DUST3965R ist eine Datenübertragungsbaugruppe DF42 mit dem entsprechenden Protokolltreiber erforderlich.

Die DUST3965R-Geräteemulation ermöglicht per Parametrierung auch eine Kommunikation über TCP/IP. Diese bietet sich z.B. dann an, wenn zwei M2000 Systeme miteinander kommunizieren sollen.

In diesem Fall wird im ORG eine DUST3965R generiert. Die Geräteemulation setzt diese Kommunikation dann auf die Windows-Socket-Schnittstelle (Winsocket) um. Anstelle der DF42 wird ein LAN/RAS Netzwerk-Adapter gemäß Windows Hardware Kompatibilitätsliste verwendet.

© IPKS GmbH 2007

DUST3966

Emulation einer Rechnerkopplungsschnittstelle DUST3966. Die DUST3966 Emulation bearbeitet die Übertragungsprotokolle HDLC U/P, HDLC U/S und HDLC B.

Damit sind beispielsweise folgende Kopplungen realisierbar:

- M2000 - M2000 über HDLC B und HDLC U und SINEC
- M2000 - IBM über SDLC und SINEC SNSNA
- M2000 - X25 Netz über HDLC B und SINEC SNPV
- M2000 über TCP/IP im LAN oder über RAS.

Nicht implementiert sind die WIKOP-Funktionen (Remote-Laden...).

Für die Emulation der DUST3966 ist eine Datenübertragungsbaugruppe DF32 oder DF42 mit dem entsprechenden Protokolltreiber erforderlich.

Die DUST3966-Geräteemulation ermöglicht per Parametrierung auch eine Kommunikation über TCP/IP. Diese bietet sich z.B. dann an, wenn zwei M2000 Systeme miteinander kommunizieren sollen.
In diesem Fall wird im ORG eine DUST3966 generiert. Die Geräteemulation setzt diese Kommunikation dann auf die Windows-Socket-Schnittstelle (Winsocket) um. Anstelle der DF32/42 wird ein Netzwerk-Adapter gemäß Windows Hardware Kompatibilitätsliste verwendet.

KS100

Emulation einer SINEC H1 Kopplung über die ISO-Protokolle bis einschließlich der Ebene 4.
Für diese Geräteemulation muss der folgende Kommunikationsprozessor unter Windows installiert sein:

- CP1413/CP1613 mit dem Softwarepaket SIMATIC NET

CP1400

Emulation einer SINEC H1 Kopplung über die ISO-Protokolle bis einschließlich der Ebene 7. Die Ebenen 5 bis 7 entsprechen dem Automatisierungsprotokoll SINEC AP 1.0 .

CS275

Anschluss eines emulierten SICOMP Systems an das TELEPERM M Bussystem CS275. Für den Einsatz des M2000 Geräteprozesses ist eine spezielle Siemens Baugruppe (NAT) notwendig,

ETC M

PromeaNET (*)
PromeaNET übernimmt die Funktion vergleichbar der SINEC ETC M für Datensichtgeräte und Drucker, wobei als Sichtgeräte die ETC M kompatiblen Terminal-Emulationen DEnet75(*) als DISIT-Terminal und DSnet75(*) als DS075-Sichtgerät auf PCs unter MS-DOS und Windows 3.1 einsetzbar sind.

Mit SPOOL75(*) funktioniert ein solches Terminal gleichzeitig als Druckerserver für seriell angeschlossenen Drucker aus der SICOMP-Reihe wie z.B. DR216N.

Vorteile und Besonderheiten:
- Jeder generierte DSSK- und DRUA-Kanal kann ohne Neugenerierung des ORG dem PromeaNET zugeordnet werden.
- 129 Kanäle sind theoretisch zu vereinbaren.
- Ein emuliertes Terminal kann mehrere Emulationen bedienen, auch gemischt mit SICOMP M, wenn diese mit einer ETC M ausgerüstet sind.

(*) Produkte der Fa. SIG Aachen

Die Promea NET Funktion ist als Produkt der Fa. SIG Aachen nicht Bestandteil des Gewährleistungs- bzw. Support-Umfanges von M2000. Eine entsprechende Unterstützung muss dort direkt angefordert werden.
IPKS reicht Problemmeldungen weiter.

© IPKS GmbH 2007

Dongle

IPKS liefert verschiedene Dongle-Typen:

- NormalDongle (mit Optionen nach Kundenbestellung) für den Dauerbetrieb in 2 Varianten: 25pol. Stecker/Buchse, USB-Stecker
- EmergencyDongle = NormalDongle für zeitlich begrenzten Betrieb der Emulation (max. 30 Tage)
- EvaluierungsDongle (mit Optionen nach Kundenbestellung) zum Testbetrieb der Emulation

Emergency-Dongle

In einem Emergency-Dongle ist die Gültigkeitsdauer (max. 30 Tage) vermerkt. Erkennt der Emulator beim Start einen 'jungfräulichen' Emergency-Dongle, wird die Meldung (Warnung) ' *Inaktiver E-Dongle*' erzeugt. Wird der Emulator innerhalb von 30 Minuten wieder beendet, bleibt der E-Dongle 'jungfräulich'.
Nach mehr als 30 Minuten Emulator-Laufzeit wird der Dongle 'aktiv', der Emulator errechnet den letzten Gültigkeitstag und meldet ' *E-Dongle bis TT/MM/JJJJ*'.
Erkennt der Emulator beim Start einen 'aktiven' und noch gültigen E-Dongle, wird die Meldung ' *E-Dongle bis TT/MM/JJJJ'* erzeugt.
Erkennt der Emulator beim Start einen 'aktiven' , aber ungültigen E-Dongle, wird die Meldung ' *E-Dongle ist abgelaufen*' erzeugt.
Am letzten Gültigkeitstag wird stündlich ein PopUp-Fenster mit der Warnung:
'*E-Dongle läuft heute ab!!! '* ausgegeben.

Wird der Gültigkeitszeitraum des Emergency-Dongle während des Emulatorbetriebs überschritten, wird der Emulator n i c h t beendet.

Evaluierungs-Dongle

Der Evaluierungs-Dongle meldet nach einer ununterbrochenen Emulator-Laufzeit von einer Stunde den Emulator an, nach einer weiteren Stunde wird der Emulator ohne zusätzliche Warnung beendet.
Ein Evaluierungs-Dongle kann beliebig oft, immer nur für 2 Stunden, eingesetzt werden.
Auch in einem Evaluierungs-Dongle müssen die benötigten Optionen freigeschaltet werden.

Windows

Der Systemname Windows steht für die Microsoft Betriebssysteme - Windows2000 / -2003Server und Windows XP.

© IPKS GmbH 2007

3 Inbetriebnahme

3.1 Lieferumfang

Lieferumfang (M2000) Im Lieferumfang M2000 sind enthalten:

- CompactDisc mit der M2000/PCSIC-M/R Software
- ein Dongle
- Handbuch auf CD

Dateien Die **CD** enthält folgende, wichtige Dateien:

readme_M2.wri	README File für Neuerungen, Tips und Tricks
M2000.pdf	M2000 Dokumentation
Eula_D.rtf	EndUser Lizenz in Deutsch
Eula_US.rtf	EndUser Lizenz in Englisch
Version.txt	Angaben zu den Versionen auf der CD

Verzeichnis *Utils*

mcsave.exe	Magnetband(MK82) Programm
pc_rdisk.exe	SICOMP Platten von MODisk kopieren
xdong.exe	Dongle-Daten ausgeben
diagnostix.exe	Aladdin SystemDatenSammler

Verzeichnis *EmulationDoc*** **M2000 Dokumentationen**
Verzeichnis *Remote_Dongle* **Treiber NetzwerkDongle + Dokumentation**
Verzeichnis *Source* **Ersatzteillager**

M2000\binm	nicht gepackt
M2000\binr	nicht gepackt
M2000\Desktop	nicht gepackt
utils	nicht gepackt

Verzeichnis *M2000* Die **M2000-Installation** speichert folgende Verzeichnisse:

Desktop	**M2000 Programme**
M2000.exe	zentrale Anwendung
M2000hsp.exe	M2000 HSP-Viewer
M2000off.exe	NotausStart-Prozess
MPseudo.exe	DialogProgramm MPseudo
M2000.dll	Library Schnittstelle M2Oberfläche-Emulation
M2000Int.dll	Library M2Oberfläche
M2000hsp.dll	Library HSP-Viewer
readme_m.wri	Aktuelle Infos zu SIC-M
readme_r.wri	Aktuelle Infos zu SIC-R
readme_M2.wri	Aktuelle Infos zu M2000
Help	M2000 Online-Hilfe
Sicview	SICOMP Festplatten Viewer
TrcView	Programm TraceView
Wmf_ico	interne M2000 Icons

Doc	**M2000 Dokumentationen**
Binm	**Bausteine SIC-M Emulation**
Binr	**Bausteine SIC-R Emulation**

Samples	**C++ Beispiele f. PSD/PSxKOMM**
Inc	Include Files
Lib	Libraries
Psexampl	Beispiele ORG/DVS Aufrufe über PSAPI
Psxkomm	Beispiele/Struktogramme PipeVerkehr ohne PSAPI

Utils* **Hilfsprogramme, DLL's**
u.a. *sickoor, mimdrv, NDIS2000, mcsave, daytimed, ...*

 © IPKS GmbH 2007

Die Anwender-Installation speichert folgende Verzeichnisse:

Sicomp	**ParameterFiles/Batches , Assembler Beipiele f. PSD** *du0x.par, emu_strt.bat, mpar.sys/rpar.sys, plsk.psd, setall.bat, spox.par*
Gsb	*bench, korxxx, koryyy, pccopy*
Kor	*korxxx, koryyy*
Psd	Beispiele f. PSD (Quellen)
Pso	Beispiele f. PSD (ObjCode)
Trc	*dsskx.tko, .trc, .par* Files

Die Dateien *mpar.sys / rpar.sys* und *dsskx.tko* können an das jeweilige System angepasst werden. Bei einem Update sollte man darauf achten, diese Dateien mit der neuen Installation nicht zu überschreiben. Sie können diese Dateien umbenennen und ihnen systemspezifische Namen geben.

Beispiel mpar.sys → audi.sys
dsskx.tko → dssk2.tko

Anmerkung Die Datei *plsk.psd* wird als SICOMP Platte geliefert, das heißt, wenn Elemente oder Programme aus den Bibliotheken der "Platte" geladen oder transferiert werden sollen, muss *plsk.psd* als logisches Plattenlaufwerk in *mpar.sys /rpar.sys* eingetragen werden.

© IPKS GmbH 2007

Lieferumfang (SIC-M) Die M2000-Installation speichert für SIC-M folgendes Verzeichnis:

Verzeichnis ***Binm***	**ZE- / Geräteemulationen**
sic_m.exe	Initialisierungsprozess
sic_ze.exe	ZE-Emulation
sic_zeit.exe	Geräteemulation Zeitgeber
sic_disk.exe	Geräteemulation Platten
sic_op31.exe	Geräteemulation optische Platte (OP31)
sic_eads.exe	Geräteemulation serielle Schnittstellen
sic_eadr.exe	Geräteemulation Drucker
sic_eamx.exe	Geräteemulation Promea MX
sic_eaes.exe	Geräteemulation ES Terminal
sic_tape.exe	Geräteemulation Magnetbandkassette
sic_flop.exe	Geräteemulation Diskettenlaufwerk
sic_du03.exe	Geräteemulation DU03
sic_du04.exe	Geräteemulation DU04
sic_du05.exe	Geräteemulation DU05
sic_du06.exe	Geräteemulation DU06
sic_du78.exe	Geräteemulation DS078
sic_rb06.exe	Remote-Boot-Funktion
sic_rs06.exe	Remote-Boot-Funktion
sic_ucp2.exe	Geräteemulation TCP/IP über UCP2
sic_ftnt.exe	Geräteemulation Pseudogerät
sic_ks09.exe	Geräteemulation KS100
sic_ks10.exe	Geräteemulation KS100
sic_ks11.exe	Geräteemulation KS100
sic_disi.exe	Anschluss DISIT-Terminal
sic_coro.exe	Anschluss VD2000
sic_etcm.exe	Anschluss SIG-Terminal-Emulationen über ETC M
sic_cp14.exe	SINEC H1 Kopplung über ISO
sic_cs27.exe	Anschluss an TELEPERM M
sic_spox.exe	Druckausgabe in eine Windows Datei

Lieferumfang (SIC-R) Die M2000-Installation speichert für SIC-R folgendes Verzeichnis:

Verzeichnis ***Binr***	**ZE- / Geräteemulationen, Psf3600.dll**
sic_r.exe	Initialisierungsprozess
sir_ze.exe	ZE-Emulation
sir_zeit.exe	Geräteemulation Zeitgeber
sir_disk.exe	Geräteemulation Platten
sir_eads.exe	Geräteemulation serielle Schnittstellen
sir_eadr.exe	Geräteemulation Drucker
sir_eaes.exe	Geräteemulation ES-Terminal
sir_eamx.exe	Geräteemulation Promea MX
sir_flop.exe	Geräteemulation Diskettenlaufwerk
sir_3962.exe	Geräteemulation DUST3962
sir_3964.exe	Geräteemulation DUST3964R
sir_39640.exe	Geräteemulation DUST3964
sir_3965.exe	Geräteemulation DUST3965
sir_3966.exe	Geräteemulation DUST3966
sir_61gw.exe	Geräteemulation 3961 Gateway
sir_cs27.exe	Anschluss an TELEPERM
sir_rb66.exe	Remote-Boot-Funktion
sir_rs66.exe	Remote-Boot-Funktion
SIR_FTN.exe	Geräteemulation Pseudogerät
sir_ftnt.exe	Geräteemulation Pseudogerät
sir_wtf.exe	Wartungsfeld
sir_xpsa.exe	Geräteemulation PSA
Psf3600.dll	

 © IPKS GmbH 2007

3.2 Systemvoraussetzungen

Für den Betrieb von M2000 muss als ' Ursprungsrechner ' ein Siemens SICOMP M/R System mit MKSK vorhanden sein. Von diesem System werden die System- und Datenplatten auf Magnetband gesichert und als Windows Dateien auf dem PC abgelegt.
Die PC Konfiguration besteht aus folgenden Hardware-Komponenten:

Prozessor Der PC muss mit einem Intel i486 bzw. Pentium Prozessor (oder kompatibler) ausgestattet sein. Obwohl Windows auch auf Hardware-Plattformen ablauffähig ist, die nicht der Intel-Prozessorarchitektur entsprechen, ist M2000 nur auf Intel-kompatiblen Prozessoren einsetzbar. Aus Performancegründen werden die SICOMP M/R Befehle direkt auf den Intel-Befehlssatz abgebildet.

Wichtig! Der interne ProzessorCache darf nicht ausgeschaltet sein, da sich sonst die Bearbeitungszeiten der Emulation um den Faktor 5 erhöhen. Ein externer Cache verbessert das Zeitverhalten zusätzlich.
Der Speicherausbau muss mindestens ≥ 64 MB betragen. Es muss gewährleistet sein, dass alle M2000 Komponenten im RAM ablaufen und nicht in die Windows Page-Datei ausgelagert werden

Grafikkarte und **Tastatur** Entsprechend den Windows-Anforderungen

Hard-Disk Da der Zugriff über das Windows Dateisystem erfolgt, muss lediglich die Kapazität der Festplatte so bemessen sein, dass die Datenplatten vollständig innerhalb einer Partition abgelegt werden können. Verschiedene PLSK können auf das gleiche oder auch auf verschiedene logische Windows Laufwerk(e) abgelegt werden.

Serielle Schnittstellen Für den Anschluss von Terminals müssen serielle Schnittstellen entsprechend den Windows Anforderungen vorhanden sein.

Terminal Für die Bedienung des emulierten SICOMP M/R-Systems können wahlweise

- Sichtgeräte über serielle Schnittstellen angeschlossen werden
  (lokal und remote, siehe Kapitel: Schnittstellenkonzentrator), d.h. es können die vorhandenen Sichtgeräte weiterhin benutzt werden.
- Terminal-Emulationen lokal, d.h. auf dem Emulationssystem mit der Windows Konsole betrieben werden. Diese können dann auch u.a. als virtuelle Konsole, Standardmelde- bzw. Standardbediengerät verwendet werden.
- Terminal-Emulationen über serielle Schnittstelle oder Netzwerk angeschlossen werden.

Drucker Anschließbar sind alle Drucker mit Centronics-Schnittstelle, entsprechend den Erfordernissen der Anwendung. Ebenso sind DR202-Drucker (auch 3918, 3919) über serielle Schnittstellen betreibbar.
(lokal und remote, siehe Kapitel: Schnittstellenkonzentrator).
Es sollten nur Drucker verwendet werden, die standardmäßig im ASCII-Modus mit eigenem Zeichensatz betrieben werden können.

Streamer Die in M2000 enthaltene MKSK Emulation unterstützt die Streamer Tandberg Data Panther TDC 3660 bzw. TDC 3820.

Floppy-Disk 3,5", 1.44 MB bzw. 5,25" 1.2 MB
(es wird die SICOMP M/R - Formatierung verwendet).

Windows Auf dem PC muss Windows 2000/2003(Server), WindowsXP installiert sein.
Die M2000 Installation unter WindowsXP erfordert ein NTFS Dateisystem.

3.3 Installation

3.3.1 Übersicht

Die Installation eines SICOMP M/R-Systems mit M2000 unter Windows auf einem PC, der die genannten Voraussetzungen (siehe Kapitel: Systemvoraussetzungen) erfüllt, besteht aus den nachfolgend beschriebenen Schritten.

Schritt 1 Der mitgelieferte StandardDongle muss auf die parallele Drucker-Schnittstelle (LPT1) gesteckt, der USB-Dongle in einen freien USB-Slot gesteckt werden und für die Laufzeit des Emulators dort verbleiben. Ohne Dongle ist die Software nicht ablauffähig.
Falls an LPT1 ein Drucker betrieben werden soll, muss das Druckerkabel am StandardDongle gesteckt sein. Der USB-Dongle sollte nicht in einen USB-Hub, sondern in einen freien USB-Slot direkt gesteckt werden.

Schritt 2 Wird ein USB-Dongle eingesetzt, muss zuerst der USB-Dongle Treiber installiert werden (**Setup – Produktinstallation – USB-Dongle Treiberinstallation**).

Schritt 3 **M2000 (incl. Emulator) installieren** (siehe 3.3.2 M2000 installieren)

Schritt 4 **Anwender-System installieren** (siehe 3.3.2.1 Anwendersystem installieren)

Schritt 5 **Kopie des SICOMP M/R Anwender-Systems installieren** (siehe OnlineHelp)

Die auf dem Ursprungssystem erzeugten Magnetbandkopien bzw. OPxx-Datenträger müssen eingelesen und als Windows-Dateien abgelegt werden. Diese Windows-Dateien stellen dann die Datenträger des SICOMP M/R Systems dar. Das Einlesen kann mit der M2000-Funktion `Daten importieren` im Register `Allgemein` – Button `Kopieren` angestoßen werden.

Alternative: Die Kopien (MKxx/Opxx) der Platten des Ursprungssystems können auch vor der Installation von M2000 mit den Programmen *mcsave.exe* bzw. *pc_rdisk.exe* (auf der CD im Verzeichnis *Utils*) übertragen werden.

Schritt 6 **Konfigurations-Datei für Anwendersystem editieren** (siehe 3.3.3 und 4.)

Für die zu emulierende Gerätekonfiguration benötigt M2000 eine Konfigurationsdatei. In dieser Datei wird das Ursprungssystem beschrieben.
In %M2000Disk%\sicomp\... werden die Beispielkonfigurationen *mpar.sys/rpar.sys* mitgeliefert. Diese (MEDISkompatiblen) Files müssen vom Anwender evtl. noch ergänzt und an seine Systemkonfiguration angepasst werden. Außerdem kann mit der M2000-Funktion `Automatisch generieren` im Register `Allgmein` eine Konfiguration erstellt werden, die dann vom M2000internen Editor zur Kontrolle vorgelegt wird und vom Anwender angepasst werden kann.

Schritt 7 **Emulation starten** (siehe 3.4)

Nachdem alle Vorbereitungen getroffen sind, können Sie mit M2000 im Menu bzw. Register `Monitor` Ihre SICOMP M/R Emulation starten.

Terminalemulation Bei der Installation von M2000 wird im Unterverzeichnis .....*Ipks**Utils*\... die aktuelle Terminalemulation (*ds074.exe bzw. te2000.exe*) abgelegt. Sie können Terminalemulationen über ein Rechnernetz direkt aus diesem Verzeichnis installieren. Die Terminalemulationen können aber nur betrieben werden, wenn Sie entsprechende Lizenzen erworben haben (Eintrag im M2000 Dongle oder separater Terminaldongle).

 © IPKS GmbH 2007

Hinweise !

Die Funktionen von M2000 sind in der OnlineHelp (`Hilfe-Hilfethemen` F1) von M2000 beschrieben.

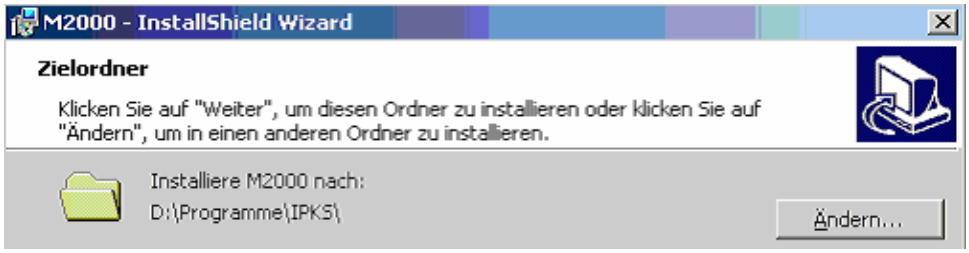
Sind keine externen USB-Slots frei, dann könnte ein USB 2.0-PCI-Hostadapter nützlich sein. Diese PCI-Karten haben mehrere, externe USB-Slots und meist auch einen internen Slot, in den der USB-Dongle gesteckt werden kann. Intern gesteckt wäre der Dongle auch vor unbeabsichtigtem oder beabsichtigtem Entfernen sicher.

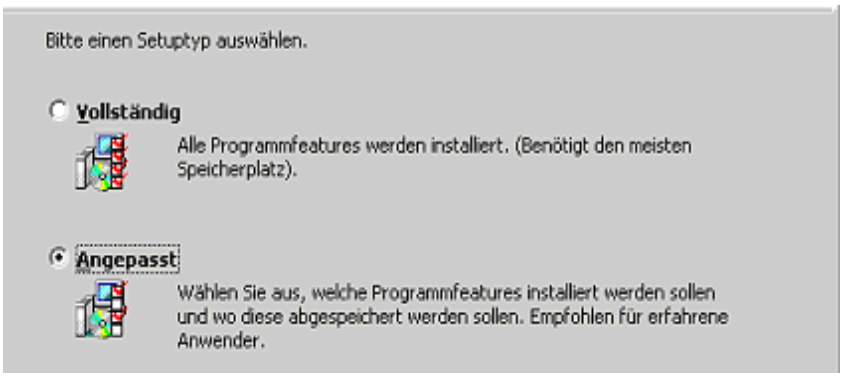
Wichtig! Auf der CD befindet sich ein README-File (*readme_M2.wri*); es enthält zusätzliche und aktuelle Informationen, die unbedingt zu beachten sind.

3.3.2 M2000 (incl. Emulator) installieren

1. Loggen Sie sich unter Windows als Administrator ein.

2. Die Installations-Routine wird nach Einlegen der CD automatisch (*autorun*) gestartet; sie (*InstalShield*) kann aber auch vom Anwender aktiviert werden (im *root*Verzeichnis der CD das Programm *setup.exe* aktivieren). Ziel der Installation ist (default) *%SystemRoot%\programme\IPKS\M2000\...*, über den Button `Ändern` kann der Zielpfad selbst bestimmt werden.
   Der SetupTyp • **Vollständig** installiert alle M2000 Features, der Typ • **Angepasst** lässt Ihnen die Wahl. Alles weitere ist selbsterklärend.


M2000 - InstallShield Wizard
Zielordner
Klicken Sie auf "Weiter", um diesen Ordner zu installieren oder klicken Sie auf "Ändern", um in einen anderen Ordner zu installieren.
Installiere M2000 nach:
D:\Programme\IPKS\
Ändern...


Bitte einen Setuptyp auswählen.
Vollständig
Alle Programmfeatures werden installiert. (Benötigt den meisten Speicherplatz).
Angepasst
Wählen Sie aus, welche Programmfeatures installiert werden sollen und wo diese abgespeichert werden sollen. Empfohlen für erfahrene Anwender.

3. Update Installation

   Eine Update-Installation erfolgt, nach Einlegen der (Update)CD und Start der (M2000)SetupRoutine, automatisch; d. h. der Anwender kann nur noch Fortgang oder Abbruch der (Update)Installation beeinflussen, er hat aber keine Möglichkeit mehr, den Installationsprozess zu steuern.
   (`Verzeichnisse ändern, Komponentenauswahl etc.`)
   Eine weitere Möglichkeit des Update bietet die Windows Systemfunktion - Start-Einstellungen-Systemsteuerung-Software - Ändern/Entfernen

   Für die möglichen Funktionen
   ⊙ Programm ändern
   ⊙ Programm reparieren
   muss unbedingt der Original-Installations-Datenträger zur Verfügung stehen.
   Alles weitere ist selbsterklärend.

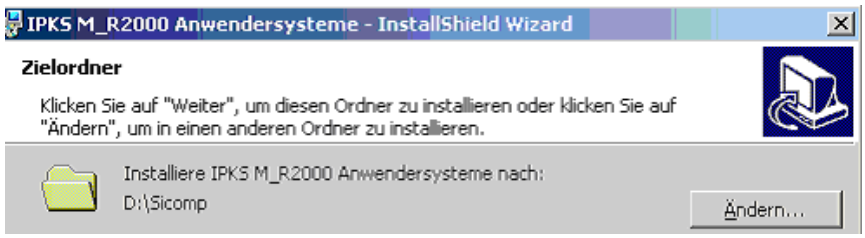
4. Nach einem RESET des PC (Start-Beenden-Neu starten) kann M2000 aktiviert und können weitere Vorbereitungen getroffen werden. (siehe 3.3.1, Schritt 3 und 4)

© IPKS GmbH 2007

3.3.2.1 Anwender-System installieren

1. Sind Sie noch unter Windows als Administrator angemeldet?

2. Über den Button Anwender-System der M2000 Produkt-Installation werden die Daten des Anwendersystems auf die Platte transferiert. Alles weitere ist selbsterklärend. Bei der Standard-Installation (• **Vollständig**) wird entweder automatisch nach *%M2000disk%\Sicomp* installiert oder Sie geben nach **Klick** auf den Button **Ändern...** ein anderes Ziel vor; bei der Anwender-spezifischen Installation (• **Angepasst**) kann auch über den Button **Ändern...** ein anderes Ziel-Verzeichnis angegeben werden.


IPKS M_R2000 Anwendersysteme - InstallShield Wizard
Zielordner
Klicken Sie auf "Weiter", um diesen Ordner zu installieren oder klicken Sie auf "Ändern", um in einen anderen Ordner zu installieren.
Installiere IPKS M_R2000 Anwendersysteme nach:
D:\Sicomp
Ändern...

Anwendersystem - InstallShield Wizard
Angepasstes Setup
Wählen Sie die Features, die Sie installieren wollen.
Klicken Sie auf eins der Symbole in der Liste, um die Installation zu ergänzen.
Anwendersystem
Feature-Beschreibung
Hilfsprogramme zur Generierung eines emulierten Sicomp-Systems aus einem Originalsystem.

3. Das Anwender-System, d. h. Parameterfiles wie *mpar/rpar.sys*, *emu_strt.bat*, Hilfsprogramme in *plsk.psd* etc., wird in %M2000dir% (z. B. C:\Sicomp\) gespeichert. Kopieren der SICOMP-Datenträger nach %M2000dir%\Platten\ siehe: 3.3.1 Übersicht, Schritt 4

3.3.2.2 Änderung von SystemOptionen

Nach der Installation von M2000 sind folgende Systemoptionen geändert:

- - Systemeigenschaften – Erweitert – Systemleistungsoptionen -
  `Systemleistung optimieren für Hintergrunddienste`
  (Default: `.... für Anwendungen`)

- – Ereignisanzeige – Anwendungs-/ Systemprotokoll – Eigenschaften -
  wenn maximale Protokollgröße erreicht, dann:
  `Ereignisse nach Bedarf überschreiben`
  (Default: `Ereignisse überschreiben, die älter als ...`)

- - Eigenschaften von Anzeige – Darstellung - Effekte -
  ☐ `Fensterinhalt beim Ziehen anzeigen` - wird deaktiviert.
  (Default: ☑ `Fensterinhalt beim Ziehen anzeigen`)

- AutoUpdate-Funktion in Registry von Windows2000SP3/WindowsXP wird abgeschaltet.
  (HKEY LocalMachine\System\CurrentControlSet\Services\wuauserv ▸ Start:)

3.3.3 Konfigurationsdatei (mpar.sys/rpar.sys)

In der Konfigurationsdatei werden Angaben über den zu emulierenden Rechner und Zuordnungen über die verwendeten Geräte gemacht. Zum Lieferumfang gehören Beispieldateien, die als Vorlage für eine individuelle Konfiguration dienen können. Die Dateien können über das M2000 Register `Monitor` im Bereich `Konfiguration` Button `Neu..` bzw. `Ändern..` einer Konfiguration zugeordnet und editiert werden. Weitere Informationen können der M2000 OnlineHelp entnommen werden.

Eine Konfigurationsdatei ist zeilenweise aufgebaut (Textdatei). Jeder Eintrag wird in einer eigenen Zeile notiert, wobei eine Zeile max. 250 Zeichen lang sein darf. Die mitausgelieferten Beispieldateien sind auf eine Zeilenbreite von 80 Zeichen formatiert und enthalten keine Umlaute; die Dateien können auch mit MEDIS editiert werden.

Jede Zeile (der ParameterTeil) sollte mit einem Semikolon abgeschlossen werden. Mit Semikolon wird der KommentarTeil der Zeile eingeleitet; dieser Teil der Zeile wird vom Emulator überlesen. Leerzeilen sind zulässig. Zwischen großen und kleinen Buchstaben wird nicht unterschieden. Jeder Eintrag beginnt mit einem Codewort, gefolgt von einem oder mehreren Parametern. Die Einträge sollten nach Möglichkeit

- mit device-Nr. aufsteigend ... (*ioadr-devnr* bzw. *anr,gnr*)
- bei PlattenLaufwerken müssen *ioadr-device*Nr. aufsteigend editiert werden
- mit Semikolon abgeschlossen werden (ParameterTeil)
- mit Kommentar versehen werden

Syntax bzw. logische Fehler in den Zeilen der Konfigurationsdatei werden im Eventlog(buch) dokumentiert.

Die genaue Notation der Einträge in der Konfigurationsdatei wird in den Kapiteln 4. und 5. beschrieben.

Übersicht der allgemeinen Einträge:

CPU	CPU-Typ	(M/R)
ZESTART	ZE Start verzögern	(M)
MAXMEM	Hauptspeicherausbau	(M/R)
IMAGE	Hauptspeicherrettdatei	(M/R)
BOOT	Adresse Urladegerät und virtuelle Konsole	(M/R)
MECR	Betrieb einer MEC-28R Anschaltung	(R)
NOFPP	Gleitpunktprozessor deaktivieren	(R)

 © IPKS GmbH 2007

Einträge für Gerätedefinitionen:

Die Rangierung eines unter SICOMP M/R benutzten Gerätes zu einem Gerät des Windows-Systems erfolgt mit Gerätedefinitions-Einträgen.
Für jedes SICOMP M/R Gerät muss in der Konfigurationsdatei ein DEVICE-Eintrag vorhanden sein. Dieser hat die folgende allgemeine Form:

SIC-M

```
DEVICE = ioadr[-devnum],keyword,"pfadname"
DEVICE = ioadr[:m],keyword,"pfadname"
DEVICE = ioadr[:m],keyword,"pipename"
DEVICE = ioadr[:m],keyword,"socketnr"
```

ioadr ist die IO-Adresse des Gerätes entsprechend der ORG-Generierung (4stellige Hexadezimalzahl). Für Geräte, die eine *DeviceNummer* haben können (z.B. Plattenlaufwerke), ist optional der Parameter *devnum* angebbar. Wenn bei Geräten wie Drucker, Sichtgeräten, Promea MX, Promea ES, FTNT nicht alle 8 Kanäle/Teilgeräte (Ein-/Ausgabe Blockbetrieb, Zeichenbetrieb, virtuelle Konsole, usw.) einer IO-Adresse genutzt werden, können mehr IO-Adressen im ORG generiert werden. Mit dem Parameter *m* wird dann der (vorher im ORG generierte) Abstand zwischen zwei IO-Adressen festgelegt. Mögliche Werte für *m* sind:
0 – Aufteilung der *ioadr* in 1er Schritten `device=A380:0,... =A381:0,...`
1 – Aufteilung der *ioadr* in 2er Schritten `device=A380:1,... =A382:1,...`
3 – Aufteilung der *ioadr* in 4er Schritten `device=A380:3,... =A384:3,...`

Beispiele

DEVICE = 8200**-7**, FP023, "\sicomp\platten\plsk007";	---	-devnum
DEVICE = A378**:1**, DRSPOTS, "I:\Drucker\Nsu",30;	---	2er Schritt
DEVICE = A37A**:1**,DRSPOTS, "I:\Drucker\In",30;	---	2er Schritt
DEVICE = A100**:0**,DR202, "..\\.\COM182";	---	1er Schritt
DEVICE = A101**:0**,DR202, "..\\.\COM183";	---	1er Schritt
DEVICE = 8008,DS074, **"DSSK0"**;	---	pipename
DEVICE = A390:1,DS074NET, **"20002"**;	---	socketnr

Der Parameter *keyword* bezeichnet das Gerät unter Windows. Mit *pfadname* wird ein Windows-Device zugeordnet, *pipename* ist der Name der *named pipe* für eine lokale Terminalemulation und *socketnr* die Socket-Nr. für eine Terminalemulation über Netzwerk.

SIC-R

```
DEVICE = anr,gnr,keyword,"pfadname"
DEVICE = anr,gnr,keyword,"pipename"
DEVICE = anr,gnr,keyword,"socketnr"
```

anr ist die Anschlussstellennummer des Gerätes entsprechend der ORG-Generierung (Dezimalzahl).

gnr ist die Gerätenummer gemäß der ORG-Generierung (Dezimalzahl).

keyword Der Parameter bezeichnet das Gerät unter Windows.

pfadname Mit diesem Parameter wird ein Windows-Device zugeordnet.

pipename (eindeutiger) Name der *named pipe* für lokale Terminalemulation

socketnr Socket-Nr. für Terminalemulation über Netzwerk

Beispiele

DEVICE = 5,2, 3948, "\sicomp\platten\Rsys0.v1";
DEVICE = 1,1, DRCOM, "COM7",3915,,100
DEVICE = 1,2, DR202, "\\.\COM17";
DEVICE = 2,0, 3974RT, "DSSE1";
DEVICE = 2,1, 3974RNET, "20001";

Übersicht der definierbaren Geräte:

SIC-M

3974MT	Betrieb einer ZBE3974MT oder einer DE75 bzw. einer kompatiblen Emulation über eine serielle Schnittstelle. Betrieb einer DISIT-Emulation TE2000 DX als lokales Terminal.
3974MTNET	Betrieb einer DISIT-Emulation TE2000 DX als Terminal im Netz.
DS074	Betrieb einer Konsolen Emulation (lokales Terminal, VK).
DS075[F]LOK	Betrieb einer Alpha-Terminal Emulation TE2000 AX (Farbe) als lokales Terminal.
DS075[F]NET	Betrieb einer Alpha-Terminal Emulation TE2000 AX (Farbe) als Terminal im Netz.
DS075NET	Betrieb einer Alpha-Terminal Emulation TE2000 FG als Terminal im Netz.
DS075	Betrieb einer Alpha-Terminal Emulation TE2000 FG.
DS075[F]	Betrieb einer DS075 (Farbe) über eine serielle Schnittstelle.
DS075 DISIT	Betrieb einer DS075 DISIT über eine serielle Schnittstelle. (TE2000 DX)
DU78	TE2078 - Emulation eines DS078 Terminals.
VDU2000	Betrieb einer VDU2000 über eine serielle Schnittstelle.
PS048	Emulation eines Teilgeräts eines PS048 Plattenlaufwerks durch eine Windows Datei (32 MB).
PS049	Emulation eines Teilgeräts eines PS049 Plattenlaufwerks durch eine Windows Datei (13 MB).
FP0XX	Emulation einer Partition eines FP023, FP024, usw. (Fest)Plattenlaufwerks durch eine Windows Datei (variable Länge).
FP0XXAE	Emulation einer Partition eines FP023, FP024, usw. (Fest)Plattenlaufwerks mit ASCII-EBCDIC Konvertierung durch eine Windows Datei (variable Länge).
FLOP	Diskettenlaufwerk 3½“, nicht SICOMP M kompatibel.
OP31	Betrieb einer optischen Platte OP31. (ab M2-MSE)
ZIG1	Emulation eines Promea-Zeitgebers.
ZIG2	Emulation eines Promea-Zeitgebers (msGenauigkeit nur bei Zeit lesen).
ZIG3	Emulation eines "millisekundengenauen" Promea-Zeitgebers.
DRUA	Umleitung eines Druckers auf die parallele Schnittstelle oder in eine Windows Datei.
DR202	Betrieb eines Druckers DR202 an der seriellen Schnittstelle.
DRCOM	Betrieb aller (seriellen) Drucker an der seriellen Schnittstelle.
DRSPOOL	Druckausgabe über einen Windows Druckertreiber.
DRSPOOLF	wie DRSPOOL, zusätzlich FF(FormFeed) am Ende der Ausgabe (LaserDrucker).
DRSPOT	Druckausgabe in eine Windows Datei (nicht DOS kompatibel).
DRSPOTS	Druckausgabe in eine Windows Datei (DOS kompatibel).
DRSPOTX	Druckausgabe in eine Windows Datei (mit Parameterdatei).
DRWIN	Druckausgabe in ein Windows Fenster.
DRWINT	Druckausgabe in ein Windows Fenster mit Datum/Zeit Ausgabe in jeder Zeile.

© IPKS GmbH 2007

MK82	Betrieb eines MK82 kompatiblen Magnetbandkassettenlaufwerks Tandberg Data Panther 250SE.	
EAMX	Emulation einer Promea MX.	
EAES	Anschluss von ES200 Terminals.	
DU02	Emulation einer Rechnerkopplungsschnittstelle DU02.	(M2-LAN)
DU03	Emulation einer Rechnerkopplungsschnittstelle DU03.	(M2-LAN)
DU04	Emulation einer Rechnerkopplungsschnittstelle DU04.	(M2-WAN)
DU05	Emulation einer Rechnerkopplungsschnittstelle DU05.	(M2-WAN)
DU06	Emulation einer Rechnerkopplungsschnittstelle DU06.	(M2-WAN)
ALEM	Emulation einer Prozesseinheit PE F7 über ProfiBus.	(M2-PEPB)
KS100	Emulation einer Kommunikationssteuerung KS100.	(M2-LAN)
CS275	Anschluss an TELEPERM M	(M2-CS)
CP1400	Emulation einer SINEC H1 Kopplung (ISO 1-7)	
UCP-2	Emulation eines Kommunikationsprozessors UCP-2.	(M2-LAN)
FTNT	Pipe-Schnittstelle zum Datenaustausch zwischen SICOMP und Windows (siehe Handbuch PSD). Pipe-Schnittstelle zum Datenaustausch zwischen zwei SICOMP Systemen unter M2000 über ein Windows Netzwerk.	(M2-OPEN)
WINCC	Kommunikations-Optimierung zwischen Intern-Timern und WINCC-PSD-DLL	
ODBC	Erweiterung der Pipe-Schnittstelle zum Datenaustausch SICOMP M/R und Windows um einen ODBC-Client.	(M2-OPEN)
PSCU	Kommunikations-Kanal zwischen SCU und SICOMP Programmen.	

Hinweis!

GERÄTENAME (fett)	es handelt sich um ein **Optionspaket**, das extra bestellt werden und im Dongle freigegeben sein muss.

SIC-R

3974M	Betrieb einer ZBE3974M oder einer DE75 bzw. einer kompatiblen Emulation über eine serielle Schnittstelle.	
3974MT	Betrieb einer DISIT-Emulation TE2000 DX als lokales Terminal.	
3974MNET	Betrieb einer DISIT-Emulation TE2000 DX als Terminal im Netz.	
3974RT	Betrieb einer Alpha-Terminal Emulation TE2000 AX als lokales Terminal.	
3974RNET	Betrieb einer Alpha-Terminal Emulation TE2000 AX als Terminal im Netz.	
3974R	Betrieb einer 3974R/DS075 über eine serielle Schnittstelle.	
DS075 DISIT	Betrieb einer DS075 DISIT über eine serielle Schnittstelle.	(TE2000 DX)

3941	Emulation eines 3941 Plattenlaufwerks mit 9744 Sektoren durch eine Windows Datei.
3942	Emulation eines 3942 Plattenlaufwerks mit 48720 Sektoren durch eine Windows Datei.
3945	Emulation eines 3945 Plattenlaufwerks mit 2048/4096/8192 Sektoren durch eine Windows Datei.
3946	Emulation eines 3946 Plattenlaufwerks mit 8192 Sektoren durch eine Windows Datei.
3947	Emulation eines 3947 Plattenlaufwerks mit 15616/1034240 Sektoren durch eine Windows Datei.
3948	Emulation eines 3948 Plattenlaufwerks mit 64640/61408 Sektoren durch eine Windows Datei.
3949	Emulation eines 3949 Plattenlaufwerks mit 25856 Sektoren durch eine Windows Datei.
FLOP	Diskettenlaufwerk 3½“, nicht SICOMP R kompatibel.
ZIG1	Emulation eines Promea – Zeitgebers.
ALE1	Emulation eines Zeitgebers 3691A.
DRUA	Umleitung eines Druckers auf die parallele Schnittstelle oder in eine Windows Datei.
DR202	Betrieb eines Druckers DR202 an der seriellen Schnittstelle.
DRCOM	Betrieb aller (seriellen) Drucker an der seriellen Schnittstelle.
DRSPOOL	Druckausgabe über einen Windows Druckertreiber.
DRSPOOLF	wie DRSPOOL, zusätzlich FF(FormFeed) am Ende der Ausgabe (LaserDrucker).
DRSPOTX	Druckausgabe in eine Windows Datei (mit Parameterdatei).
DRWIN	Druckausgabe in ein Windows Fenster.
DRWINT	Druckausgabe in ein Windows Fenster mit Datum/Zeit-Ausgabe in jeder Zeile.
MK82	Betrieb eines MK82 kompatiblen Magnetbandkassettenlaufwerks Tandberg Data Panther TDC 3660, TDC 3820.
EAES	Anschluss von BDE Terminals der 38xx Familie.
3961	Emulation einer Kopplung über LAN (GatewayFunction). (M2-WAN)
3962	Emulation einer (Lichtwellenleiter)Kopplung (GatewayFunction). (M2-WAN)
3964R	Emulation einer Rechnerkopplungsschnittstelle DUST3964R. (M2-WAN)
3965	Emulation einer Rechnerkopplungsschnittstelle DUST3965 (MSV2). (M2-WAN)
3966	Emulation einer Rechnerkopplungsschnittstelle DUST3966. (M2-WAN)
FTNT	Pipe-Schnittstelle zum Datenaustausch zwischen SICOMP R und Windows. (siehe Handbuch PSD) Pipe-Schnittstelle zum Datenaustausch zwischen zwei SICOMP R Systemen unter M2000 über ein Windows Netzwerk. (M2-OPEN)
WINCC	Kommunikations-Optimierung zwischen Intern-Timern und WINCC-PSD-DLL.
EAMX1	Emulation einer SonderPromea für Eingaben an das ORG.
PE3600	Emulation einer Prozesseinheit PE3600 über Profibus Peripherie. (M2-PEPB)
ODBC	Erweiterung der Pipe-Schnittstelle zum Datenaustausch SICOMP R. und Windows um einen ODBC-Client. (M2-OPEN)
OP11	Emulation der optischen Platte OP11. (ab M2-MSE)
PSCU	Kommunikations-Kanal zwischen SCU und SICOMP-R Programmen.

Hinweis!
GERÄTENAME (fett) es handelt sich um ein **Optionspaket**, das extra bestellt werden und im Dongle freigegeben sein muss.

© IPKS GmbH 2007

3.4 Bedienung des Emulators mit M2000

Vor dem Start der Emulation müssen die Konfigurationsdateien *mpar.sys* bzw. *rpar.sys*, sowie die StartBatches *emu_strt.bat* und *setall.bat* an die Anwenderkonfiguration angepasst werden.

Starten der Emulation

Start einer SICOMP M/R Emulation über den Button **Start**, Register `Monitor` oder **Emulation** ► **Starten**, Menu `Monitor`.
siehe auch: M2000 Menu- bzw. Register-`Monitor` oder in der M2000 OnlineHilfe. (Menu `Hilfe-Hilfethemen` bzw. F1 oder `MausClickrechts` zur Kontext bezogenen DirektHilfe)

Im Emulator-Startbatch (*emu_strt.bat*) wird folgende Kommandozeile durchlaufen:

```
start "EMULATOR" /min sic_m ..\mpar.sys      (mit DOS-Fenster minimiert nach dem Start)
oder
start "EMULATOR" /min sic_r ..\rpar.sys      (mit DOS-Fenster minimiert nach dem Start)
                                  |      |
                                  |      |___ Konfigurationsdatei
                                  |___ Emulator-Init-Prozess
```

bzw.

```
sic_m ..\mpar.sys             (ohne DOS-Fenster nach dem Start)
oder
sic_r ..\rpar.sys             (ohne DOS-Fenster nach dem Start)
   |      |
   |      |___ Konfigurationsdatei
   |___ Emulator-Init-Prozess
```

Alle beteiligten (Geräte)Prozesse werden im linken Fenster des M2000Register **Ausgabe** in einem festen Format ausgegeben.
(name_ioadr Pid Options Traceflags) (M)
(name_anr,gnr Pid Options Traceflags) (R)

Mit dem Start des Emulators läuft das SICOMP-Betriebssystem an. Alle weiteren Bedienungen erfolgen in der ORG/BS-M/R Notation.

Stoppen der Emulation

siehe: M2000 Menu- bzw. Register- `Monitor` oder in der M2000 Hilfe.
Der jeweilige Zustand der Emulation wird in der M2000 Statuszeile angezeigt.
(Emulation inaktiv)
Dies entspricht einem RESET des SICOMP M/R Systems mit allen Konsequenzen, z.B. bezüglich offener DVS-Dateien.

Stoppen der ZE

siehe: M2000 Register- `Emulation` oder in der M2000 Hilfe.
Der jeweilige Zustand der Emulation wird in der M2000 Statuszeile angezeigt.
(M-ZE gestoppt)

4 Konfigurationsdatei für SIC-M

4.1 Allgemeine Einträge

CPU-Typ **CPU** = *typ*

Die Angabe des CPU-Typs *typ* ist vor allem für hardwarenahe Programme und für die Genauigkeit von Rechenergebnissen wichtig.

Folgende CPU-Typen werden unterstützt: ZE01
ZE02
ZE03

Beispiel `cpu = ze03;`

Hauptspeicherausbau **MAXMEM** = *size*

size wird als Dezimalzahl angegeben und bezeichnet die Größe des Hauptspeichers in KW im Bereich von 64 bis 4032. Dies entspricht dem Maximalausbau von 8 MB des Ursprungssystems abzüglich 64 KW Festwertspeicher. Der Parameter kann dem Generierprotokoll des Originalsystems entnommen werden.

Beispiel `maxmem = 4032;`

ZE Start verzögert **ZESTART** = *n* (Sekunden) 0 <= n <= 120

Virtuelle Adressierung **mode = virtual**

Datenaustausch zwischen ZE und Peripherie-Gerät wird immer virtuell adressiert. Muss-Parameter bei HSP > 4096KW (externer ZusatzSpeicher).

Hauptspeicher-RettDatei **IMAGE** = "*pfadname"[,mode]*

Mit diesem Parameter wird eine Windows-Datei definiert, in die M2000 beim Programmende den Hauptspeicherinhalt des emulierten SICOMP M Systems schreibt (es wird ein batteriegepufferter Hauptspeicher emuliert). Programmende heißt, dass die Emulation definiert beendet wird (Button/MenuZeile **Stoppen** in M2000).

pfadname Vollständiger Pfadname einer Windows-Datei. Sie wird implizit eingerichtet.

mode Bearbeitungsmodus für die Image Datei.

NEW (auch Default). Die Image Datei wird beim Programmende immer neu unter dem gleichen Namen angelegt. Eine bereits bestehende Image Datei wird zuvor gelöscht.

ALTx Mit der Angabe ALTx (x=1 bis 9) werden x Image Dateien angelegt, die alternierend überschrieben werden. Die Image Dateien haben die Namen *dateiname.im1....dateiname.im9*. Im Kopf jeder Image Datei werden Datum/Uhrzeit und eine Konsistenzkennung hinterlegt.
Bei Beginn des Schreibvorgangs wird eine Inkonsistenzkennung eingetragen, wenn der HSP Inhalt vollständig abgespeichert ist, wird eine Konsistenzkennung mit Datum/Uhrzeit gesetzt. Überschrieben wird immer die älteste oder eine inkonsistente Datei.

Wird die Emulation wieder gestartet und die Hauptspeicher RettDatei ist vorhanden, so wird der Hauptspeicher des emulierten SICOMP M Systems aus der Hauptspeicher RettDatei vorbelegt. Dem ORG werden die entsprechenden Kennungen für einen batteriegepufferten Wiederanlauf übergeben. Damit kann eine Rekonstruktion von DVS-Dateien mittels DFCONS erfolgen.

Beispiel `IMAGE = "\sicomp\platten\sysrett.v1"`

 © IPKS GmbH 2007

PopUp Fenster mit Hinweis auf interne Probleme

mode = popup

Wird ausführlich im EventLog dokumentiert.

Urladegerät und virtuelle Konsole

BOOT = *hexboot,hexvc*

Mit diesem Parameter wird das *boot_device* festgelegt und die virtuelle Konsole bestimmt.

hexboot IO-Adresse des *boot_device* als 4stellige Hexadezimalzahl. Eine zusätzliche Teilgerätekennung wird wie folgt in die IO-Adresse integriert:

DeviceNr.	-	TeilgeräteNr.	ergibt	Bootdevice
8100	**-**	**1**		**8110**

hexvc IO-Adresse der virtuellen Konsole

Beispiel

```
BOOT = 8110, 8008;
```

4.2 Einträge für Gerätedefinitionen

4.2.1 Sichtgeräte

Sichtgeräte seriell angeschlossen

DEVICE = *ioadr,keyword,"pfadname"*[,*upar*[,*kennung*[,*timer*]]]

Mit diesem Parameter wird der Anschluss eines Sichtgerätes über eine serielle Schnittstelle (COMxx) definiert.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung
(4stellige Hexazahl)

keyword Folgende Geräte können hier angegeben werden:
DS075 oder DS075SER
DS075F oder DS075FSER
3974MTSER
3974MTFSER
DS075_DISIT
DU78
VDU2000

Für die Terminal-Emulationen muss *keyword* wie folgt verwendet werden:

DS075, DS075SER	für TE2000 AX bzw. FG oder TE-ALPHA
DS075F, DS075FSER	für TE2000 AX oder TE-ALPHA (Farbe)
3974MTSER	für TE2000 DX oder TE-DISIT
3974MTFSER	für TE2000 DX oder TE-DISIT (Farbe)
DS075_DISIT	für TE2000 DX oder TE-DISIT (Farbe)
DU78	für TE2078
VDU2000	für VDU2000

pfadname Bezeichnung der unter Windows definierten seriellen Schnittstelle. Der *pfadname* wird in der Form "COMx" bzw. "\\.\COMxx" notiert.
Für DS075 kann die serielle Schnittstelle auch auf einem entfernten Rechner sein. Der *pfadname* wird in der Form “REM:x“ angegeben. Der Parameter x steht für die Portnummer des ersten (von zwei) TCP/IP-Sockets, den der Geräteprozess auf dem M2000 Rechner einrichtet (siehe Kapitel: Schnittstellenkonzentrator).

upar Modus 1: nur für DISIT Terminal-Emulationen: Überparametrierung (Hexa)
= 0: Überparametrierung möglich
= 1: Schnittstelle vor Überparametrierung schützen (default)
Modus 2: Unklar-Erkennung (Hexa)
= 100: Unklar-Erkennung bei TTY Schnittstelle
(Brücke zwischen RxD und CTS erforderlich)
= 200: DSR Abfrage vor Ausgabe auf V.24 Schnittstelle

Die Modi 1 und 2 können mit 'ODER' verknüpft werden

kennung Freigabe der Terminal-Emulation im externen/internen Dongle
Kennung **IPKS**: Funktionsfreigabe der (alten) TE-ALPHA/TE-DISIT Terminal-Emulation im zentralen M2000 Dongle.
Kennung **TE2000 / TE2000FG**: Funktionsfreigabe der (neuen) TE2000AX/DX/FG Terminal-Emulation im zentralen M2000 Dongle.
Kennung **NONE**: Funktionsfreigabe im externen Dongle erforderlich.

timer Zeit in ms um Ein-/Ausgaben bei langsamen Sichtgeräten (z.B. ES05) oder Anwenderprogrammen zu verzögern.

 © IPKS GmbH 2007

Hinweise

- Die Schnittstelle ist vorher mit dem Windows *mode*-Kommando entsprechend zu parametrieren.
- Die Flusskontrolle wird von M2000 auf XON/XOFF=93/94h eingestellt.
- Derzeit ist nur das Überparametrieren der Baudrate möglich
- Eine ausführliche Beschreibung enthält das Benutzerhandbuch: **TE2000 Terminal-Emulation**
- Siehe auch: Kapitel 4.2.1 Sichtgeräte Tabelle 1
- Bei fehlenden Optionen in der Parameterzeile müssen Kommas trotzdem eingetragen werden.

Anmerkung

Bei einer Parametrierung über das ORG (/PARAM ..) werden vom M2000 Treiberprozess die Byte 0,1,2,3,4 ausgewertet. Bei TE2000AX/TE-ALPHA (DS075) wird zusätzlich Byte 13 ausgewertet.

Wichtig!

Die Kennungen für die Funktionsfreigabe (TE2000 bzw. IPKS) sind nicht austauschbar, d.h. sie können nur mit der jeweiligen (TE2000/TE-ALPHA/-DISIT) Emulation benutzt werden.

Beispiele

Anschluss eines Terminals an eine lokale COM Schnittstelle.
`DEVICE=A308:3,DS075,"\\.\COM25",101,IPKS,500`
Terminalanschluss (TTY) mit unklar Erkennung, Überparametrieren nicht möglich, zentrale Funktionsfreigabe, Verzögerungszeit 500 ms.

`DEVICE=A308:1,DS075SER,"\\.\COM25",200,NONE,0`
Terminalanschluss (V.24) mit unklar Erkennung und Überparametrieren möglich, externem Dongle , keine Verzögerungszeit.

Anschluss eines DS075F Terminals an eine lokale COM Schnittstelle.
`DEVICE=8008,DS075F,"\\.\COM26",,TE2000`

Anschluss eines DS075 Terminals an die COM-Schnittstelle eines entfernten Windows Systems.
`DEVICE=8008,DS075,"REM:22000"`
`DEVICE=A308:1,DS075SER,"REM:22000"`

Anschluss eines DS078 Terminals an eine lokale COM Schnittstelle.
`DEVICE=8030,DU78,"\\.\COM7";` TE2078 Terminal Emulation

Terminal-Emulation lokal

DEVICE = *ioadr*,*keyword*,"*pipename*"[,*timer*]

Mit diesem Parameter wird eine Terminal-Emulation als lokale Windows Applikation aktiviert.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

keyword Folgende Typangaben sind zulässig:

DS074 / DS074F	für eine lokale Konsole/Emulation (Farbe)
DS075LOK / DS075FLOK	für eine lokale Konsole/Emulation (Farbe)
3974MT oder 3974MTLOK	für eine lokale DISIT-Konsole/Emulation
DS075_DISITLOK	für eine lokale DISIT-Konsole/Emulation

pipename Eindeutiger Name für die Konsolen-Applikation. Unter diesem Namen wird eine *named pipe* angelegt, die damit systemweit eindeutig sein muss. Zusätzlich erscheint dieser Name als Fensterüberschrift.

timer Zeit in ms um Ein-/Ausgaben bei langsamen Sichtgeräten (z.B. ES05) oder Anwenderprogrammen zu verzögern.

Hinweise

- Bei Verwendung im Fenster sollte für den Monitor mindestens eine Auflösung von 800x600 (16/256 Farben) eingestellt sein.
- Dieser Parameter kann mehrfach definiert werden, um mehrere Sichtgeräte zu emulieren. Für jedes Gerät wird dann ein eigenes Fenster angelegt.
- Zur besseren Performance sollte eine leistungsfähige Grafikkarte (PCI) eingesetzt werden.
- Verwenden Sie in dem Pipenamen bitte keine Sonderzeichen wie: (- \ , / . ? usw.). Es kann sonst vorkommen, daß mehrere Terminalemulationen mit der gleichen Pipe arbeiten.
  Beispiel: statt A-DSSK002 schreiben Sie bitte A_DSSK002
- Der Leistungsumfang dieser Geräteemulation ist unter:
  6.1 Geräteemulation DS074 kurz beschrieben.
  Eine ausführliche Beschreibung enthält das Benutzerhandbuch:
  TE2000 Terminal-Emulation

Beispiele

```
DEVICE=8008,DS074,"DSSK0";
DEVICE=A308:3,DS074,"DSSK0";
DEVICE=8008,DS075_DISITLOK,"DSSK0";
```

 © IPKS GmbH 2007

DISIT-Emulation TE2000 DX
über Netzwerk

DEVICE = *ioadr*,*keyword*,"*socketnr*[*/t*]"[,TE2000[,*timer*]]

Mit diesem Parameter wird der Betrieb einer DISIT Terminal-Emulation TE2000 DX über Netzwerk definiert.

DISIT-Emulation TE-DISIT
über Netzwerk

DEVICE = *ioadr*,*keyword*,"*socketnr*[*/t*]"[,IPKS[,*timer*]]

Mit diesem Parameter wird der Betrieb einer (alten) DISIT Terminal-Emulation TE-DISIT über Netzwerk definiert.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

keyword Folgende Typangaben sind zulässig:

3974MTNET	TE2000 DX/TE-DISIT über Netzwerk
3974MTNETF	TE2000 DX/TE-DISIT (Farbe) über Netzwerk
DS075_DISITNET	DS075-DISIT über Netzwerk

socketnr Socket-Nummer, über die M2000 mit der Terminalemulation kommunizieren soll. Socket-(Port)Nummer, gemäß Administration.
Empfohlener Bereich: >10000
Eine aufgebaute Verbindung zwischen M2000 und der Terminalemulation wird zeitüberwacht. Wenn innerhalb von t (default 180) Sekunden kein Datenverkehr stattfindet, baut M2000 die Verbindung wieder ab. Die Überwachungszeit *t* kann mit der Socketnummer parametriert werden (siehe Beispiele).

TE2000 Funktionsfreigabe der (neuen) TE2000 DX Terminal-Emulation im zentralen M2000 Dongle.

IPKS Funktionsfreigabe der (alten) TE-DISIT Terminal-Emulation im zentralen M2000 Dongle.

timer Zeit in ms um Ein-/Ausgaben bei langsamen Sichtgeräten (z.B. ES05) oder Anwenderprogrammen zu verzögern.

Beispiele

```
DEVICE=8030,DS075_DISITNET,"20011";
DEVICE=A308:3,3974MTNET,"20011";     Zeitüberwachung default
DEVICE=A308:3,3974MTNET,"20011/3600";Zeitüberwachung 1 Std
```

DS078-Emulation TE2078
über Netzwerk

DEVICE = *ioadr*,DU78,"SOCKET",*port-o*,*host-remote:port-r*

Mit diesem Parameter wird der Betrieb einer DS078 Terminal-Emulation TE2078 über Netzwerk definiert.

ioadr wie oben

port-o(wn) Port-/SocketNummer im eigenen Rechner, über die M2000 mit der Terminalemulation (und umgekehrt) kommunizieren soll. Die Socket Nummer sollte vom Netz-Administrator vergeben werden.
Empfohlener Bereich: >10 000

host-remote Host Name oder Host Adresse im eigenen bzw. entfernten Rechner. (z.B.: Emulation-PC oder 192.168.1.111)

port-r(emote) PortNummer/SocketNummer im eigenen bzw. entfernten Rechner, über die M2000 mit der Terminalemulation (und umgekehrt) kommunizieren soll. SocketNr. *port-r* muss von SocketNr. *port-o* verschieden sein.

Beispiele

```
DEVICE=8410,DU78,"SOCKET",20000,192.168.1.49:20001;
```

Hinweis Eine ausführliche Beschreibung enthalten die Benutzerhandbücher:

TE2000/TE2078 Terminal-Emulation

Terminal-Emulation TE2000 AX über Netzwerk

DEVICE = *ioadr*,*keyword*,"*socketnr[/t]*"[,*kennung*[,*timer*]]
Mit diesem Parameter wird der Betrieb einer (neuen) Terminal-Emulation TE2000 AX über Netzwerk definiert.

Terminal-Emulation TE-ALPHA über Netzwerk

DEVICE = *ioadr*,*keyword*,"*socketnr[/t]*"[,IPKS[,*timer*]]
Mit diesem Parameter wird der Betrieb einer (alten) Terminal-Emulation TE-ALPHA über Netzwerk definiert.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

keyword Folgende Typangaben sind zulässig:
DS074NET / DS074FNET TE2000 AX/TE-Alpha über Netzwerk
DS075NET / DS075FNET TE2000 AX über Netzwerk

socketnr Socket-Nummer, über die M2000 mit der Terminalemulation kom-munizieren soll. Die Socket Nummer muss beim Start der entfernten Terminalemulation als Startparameter mit übergeben werden. Socket-Portnummer, gemäß Administration.
Empfohlener Bereich: >10000.
Eine aufgebaute Verbindung zwischen M2000 und der Terminalemulation wird zeitüberwacht. Wenn innerhalb von t (default 180) Sekunden kein Datenverkehr stattfindet, baut M2000 die Verbindung wieder ab. Die Überwachungszeit *t* kann mit der Socketnummer parametriert werden (siehe Beispiele).

kennung Freigabe der Terminal-Emulation im Dongle
Kennung **IPKS**: Funktionsfreigabe der (alten) TE-ALPHA/TE-DISIT Terminal-Emulation im zentralen M2000 Dongle.
Kennung **TE2000 / TE2000FG**: Funktionsfreigabe der (neuen) TE2000-AX/-FG Terminal-Emulation im zentralen M2000 Dongle.

timer Zeit in ms um Ein-/Ausgaben bei langsamen Sichtgeräten (z.B. ES05) oder Anwenderprogrammen zu verzögern.

Beispiele

```
DEVICE=8031,DS074FNET,"20011";
DEVICE=A301:3,DS075NET,"20011";
DEVICE=A308:3,DS074NET,"20011"      ;Zeitüberwachung default
DEVICE=A308:3,DS074NET,"20011/0"    ;keine Zeitüberwachung
DEVICE=A308:3,DS074NET,"20011/3600" ;Zeitüberwachung 1 Std
```

Tabelle 1 Sichtgerätetyp - keyword für die Parametrierung in *mpar.sys*

Sichtgeräte Typ	**keyword** serieller Anschluss	Lokales Terminal	Netzwerk Terminal
3974R	DS075	—	—
3974MT	3974MTSER	—	—
DS074/75	DS075	—	—
DS078	DU78	—	DU78
DS075_DISIT	DS075_DISIT	DS075_DISITLOK	DS075_DISITNET
TE2000 FG	DS075	DS075LOK	DS075NET
TE2000 AX	DS075	DS074/DS075LOK	DS074NET/DS075NET
TE2000 AX	DS075F	DS074F/DS075FLOK	DS074FNET/DS075FNET
TE2000 DX	3974MTSER	3974MT	3974MTNET
TE2000 DX	3974MTFSER	3974MTLOK	3974MTNETF
TE-KONS	—	DS074	—

© IPKS GmbH 2007

Anmerkung Sind im Rechner zwei (oder mehr) Netzwerkkarten installiert, kann einem Sichtgerät auch gezielt die IP-Adresse (oder der Alias-Name) der jeweiligen Karte zugeordnet werden. Der Alias-Name muss in der Datei *hosts* in **systemdir***system32**drivers**etc**...* eingetragen sein. Die zugehörige Parameterzeile ist folgendermaßen aufgebaut:

DEVICE=*ioadr*,*keyword, ipadr*:*port[/t],kennung*
DEVICE=*ioadr*,*keyword,aliasname:port[/t],kennung*

Beispiel

```
DEVICE=8030,3974MTNETF,192.168.2.10:20010/0,IPKS;
```

```
DEVICE=8030,3974MTNETF,NetCard-LAN1:20010,IPKS;
```
Verbindung wird über Netzwerkkarte 1 aufgebaut.

```
DEVICE=8038,DS075FNET,192.168.2.13:20013/0,TE2000;
```

```
DEVICE=8038,DS074NET,192.168.2.13:20013/0,IPKS;
```

```
DEVICE=8038,DS075NET,NetCard-LAN2,20013,TE2000;
```
Verbindung wird über Netzwerkkarte 2 aufgebaut.

Hinweis Eine ausführliche Beschreibung enthält das Benutzerhandbuch:

TE2000 Terminal-Emulation

4.2.2 Speichermedien

Belegmodus der Plattenlaufwerke festlegen

`mode = share`

Mit dem Parameter MODE wird der Belegungsmodus der Platten gewählt. (exclusiv/shared)
Fehlt der Parameter, ist der Belegmodus `exclusiv` eingestellt (default).
Die Verwendung des Parameters ermöglicht den Zugriff auf die Platten aus beiden Systemumgebungen.
Der Modus `shared` muss verwendet werden, wenn mit dem Windows Programm *sicview.exe* (siehe: Zusatzprogramme) auf Platten eines emulierten SICOMP Systems zugegriffen werden soll.
Der Modus kann weiterhin verwendet werden, um Platten eines emulierten SICOMP Systems unter Windows zu sichern, ohne den Emulator zu beenden. Hierbei muss aber darauf geachtet werden, dass beispielsweise DVS Dateien geschlossen sind bzw. das Anwendersystem sinnvollerweise nicht gestartet ist.

Plattenlaufwerk **PS048, PS049**

`DEVICE` = *`ioadr`*`[-`*`devnum`*`],`*`keyword`*`,"`*`pfadname`*`"[,`*`flags`*`]`

Mit diesem Parameter wird ein Teilgerät eines PS048/49-Plattenlaufwerks zur Emulation durch eine Windows Datei definiert.

`ioadr` IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

`-devnum` Devicenummer des Teil-Gerätes

`keyword` PS048 bezeichnet ein Gerät mit 64640 Sektoren
PS049 bezeichnet ein Gerät mit 25856 Sektoren

`pfadname` Pfadname der entsprechenden Windows-Datei. Sie muss immer vollständig angegeben werden. Eine nicht vorhandene Datei richtet der Emulator in der parametrierten Länge ein. Sie kann dann mit INITM formatiert und eingerichtet werden oder mit MCSAVE physikalisch beschrieben werden.

`flags` Parametrierung (Hexa)
= 1 WRITE THROUGH
Der Parameter gewährleistet, dass veränderte Daten auf die Platte geschrieben werden und nicht im CACHE verbleiben;
Option ist sinnvoll bei DVS Datenträgern.

Wichtig ! Die Parametereinträge der eingetragenen Platten müssen in der Konfigurationsdatei *mpar.sys* so angegeben werden, dass *ioadr-devnum* aufsteigend sortiert ist.

Beispiel

```
DEVICE=8200-0,PS048,"c:\sicomp\platten\SYSPS0.sic";
```


© IPKS GmbH 2007

Plattenlaufwerk **FP0xx**

DEVICE = *ioadr*[*-devnum*],*keyword*,*anzsek*,"*pfadname*"[,*flags*]

Dieser Parameter definiert eine Partition eines FP023, FP024 usw. Plattenlaufwerks zur Emulation durch eine Windows Datei.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

-devnum Devicenummer des Gerätes

keyword Gerätebezeichnung, zulässig sind: FP023, FP024, FP025, FP026, FP030, FP0XX, FP0XXAE

anzsek Anzahl der Sektoren (max. 65535)

pfadname Pfadname der entsprechenden Windows-Datei. Eine nicht vorhandene Datei richtet der Emulator in der parametrierten Länge ein. Sie kann dann mit INITM formatiert und eingerichtet werden oder mit MCSAVE physikalisch beschrieben werden.

flags Parametrierung (Hexa)
= 1 WRITE THROUGH
(siehe oben)

Wichtig! Die Parameterzeilen der eingetragenen Platten müssen in der Konfigurationsdatei *mpar.sys* so angegeben werden, dass *ioadr-devnum* aufsteigend sortiert ist.

Beispiel

```
DEVICE=8200-0,FP023,63492,"c:\sicomp\platten\DVSPS0.sic",1;
```

Umcodierung von ASCII nach EBCDIC und umgekehrt

Wenn eine Platte als FP0XXAE parametriert ist, dann wird vor dem Schreiben auf Platte geprüft, ob im PHYSPB-Wort 3 (ADDIN1) das Bit 14 (SIBI) gesetzt ist. Wenn ja, dann werden die zu schreibenden Daten von ASCII nach EBCDIC umcodiert; genauso wird nach dem Lesen von Platte geprüft, ob im PHYSPB-Wort 3 (ADDIN1) das Bit 14 (SIBI) gesetzt ist. Wenn ja, dann werden die eingelesenen Daten von EBCDIC nach ASCII umcodiert.

Beispiel

```
DEVICE=8100-0,FP0XXAE,64640,"c:\sicomp\platten\sysPS0.sic";
```

Diskettenlaufwerk

DEVICE = *ioadr,*FLOP,"*pfadname*"
DEVICE = *ioadr,*FLOP18,"*pfadname*"

Dieser Parameter definiert ein Diskettenlaufwerk.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

FLOP Betrieb eines PC-Diskettenlaufwerks (3,5", 1.44MB oder 5.25", 1.2MB) ohne Kompatibilität zu SICOMP M. Bei einem PC, der die FM Modulation für Spur 0/Seite 0 nicht unterstützt, kann dieses Diskettenlaufwerk zur Datensicherung und zum Datenaustausch zwischen mehreren Emulationen genutzt werden. Im emulierten System wird das Diskettenlaufwerk FLOP wie ein FD044 genutzt (auch mit INITM). Eine mit FLOP beschriebene (15 Sektoren Format) Diskette ist aber auf einem SICOMP M Rechner nicht lesbar.

FLOP18 wie oben, aber im 18 Sektoren Format beschrieben.

pfadname Der Pfadname heißt: \\.\A:
\\.\B:

Beispiel

```
DEVICE=8104,FLOP,\\.\A:;
```

Magnetband-Kassetten-Laufwerk **MK82**

DEVICE = *ioadr*,MK082,"*pfadname*"

Dieser Parameter ermöglicht den Betrieb eines MK82 kompatiblen Magnetbandkassettenlaufwerks Tandberg Data Panther TDC 3660 bzw. 3820 zum Lesen und Schreiben von 150 MB-Kassetten.
Magnetbandkassetten eines MK80/81 Gerätes mit 60MB können nur gelesen werden.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

pfadname Pfadname des entsprechenden Windows-Gerätes

Wichtig! Soll das MK82 unter dem GBS verwendet werden, so muss unbedingt die **IO-Adresse 8106** verwendet werden. Gegebenenfalls müssen für das Anwendersystem und den GBS-Betrieb unterschiedliche Konfigurationsdateien (*mpar.sys*) angelegt werden.

Ein im M-ORG generiertes Magnetband-Laufwerk muss auch dann in der Datei MPAR.SYS parametriert werden, wenn am PC selbst kein Magnetband-Laufwerk angeschlossen ist. Nur in diesem Fall werden Lese- oder Schreibaufrufe an Gerät MKSKm.n mit einer Fehlermeldung abgewiesen; andernfalls interpretiert der Emulator den versuchten Transfer als Auftrag an eine Festplatte, was zu unterschiedlichen Folgefehlern führen kann.
z.B. Absturz des Plattenprozesses.
Eine Software-Lösung für dieses Problem steht nicht zur Verfügung.

Beispiel

```
DEVICE=8106,MK082,\\.\TAPE0;
```

Optische Platte **OP31**

DEVICE = *ioadr*,OP31,"*pfadname*"

Dieser Parameter definiert die Emulation einer wiederbeschreibbaren optischen Platte OP31 über das Laufwerk SONY RMO-S580.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

pfadname Der Pfadname heißt "\\.\PhysicalDrive%"
% ist die Disknummer innerhalb Windows, beginnend mit 0 (siehe Windows Festplatten Manager)

Beispiel

```
DEVICE=8202,OP31,\\.\PhysicalDrive1;
```

Hinweis Für jedes an den PC angeschlossene optische Plattenlaufwerk ist in der Parameterdatei nur eine DEVICE ... Parameterzeile einzutragen (es gibt keine Teilgeräte).

siehe auch: OP31-Emulation

 © IPKS GmbH 2007

4.2.3 Drucker

Druckerbetrieb an der **Parallel-Schnittstelle**

DEVICE = *ioadr*,DRUA,"*pfadname*"[,*flags*[,*timer*]]

Mit diesem Parameter wird das Gerät *DRUA* auf einen an der parallelen Schnittstelle installierten Drucker umgeleitet.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

pfadname Pfadname des entsprechenden Windows-Gerätes, in der Regel "LPT1"

flags Parametrierung (Hexa) = 0

timer Wert in Sek. für WRITE Überwachung (Default: 20 Sec)

Beispiel

```
DEVICE=8020,DRUA,"LPT1";
DEVICE=A300:3,DRUA,"LPT1";
```

Wichtig! Für die parallele Schnittstelle darf in Windows kein Druckertreiber installiert bzw. aktiviert sein.

Druckausgabe über **Windows Druckertreiber**

DEVICE = *ioadr*,DRSPOOL,"*pfadname*"[,timer]
DEVICE = *ioadr*,DRSPOOLF,"*pfadname*"[,timer]

Mit diesem Parameter wird das Gerät DRUA an einen Windows Druckertreiber weitergeleitet. Dabei wird mit der ersten Ausgabe ein Dokument eröffnet und dieses wieder geschlossen, wenn (default) 5 Sek. keine weitere Ausgabe erfolgt. Eine erneute Ausgabe nach mehr als (default) 5 Sek. führt zur Eröffnung eines neuen Dokuments.
DRSPOOLF schließt Ausgaben mit einem zusätzlichen Steuerzeichen FF (FormFeed) ab (bei einigen LaserDruckern erforderlich).

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pfadname Pfadname des entsprechenden Windows Geräts. (lokal oder im Netz)

timer Angabe in Millisekunden, nach denen ein Dokument wieder geschlossen wird. Ist der Parameter nicht angegeben, wird als Defaultwert 5000 ms eingestellt.

Beispiele

```
DEVICE=8020,DRSPOOL,"LPT1";
DEVICE=8020:1,DRSPOOL,"LPT1",1000;
DEVICE=8020,DRSPOOL,"\\DRSERVER\DRUA00",7000;
DEVICE=8020,DRSPOOLF,\\192.168.1.7\DRUA10;
```

Anmerkung Bei Ausgabe auf einen "NetzDrucker" bedeutet:

DRSERVER	=	Name des Druck(er)Servers
192.168.1.7	=	IP Adresse des Druck(er)Servers
DRUA00	=	lokaler Freigabename des "NetzDruckers"
DRUA10	=	lokaler Freigabename des "NetzDruckers"

Druckausgabe in **Windows Datei**

DEVICE = *ioadr*,DRUA,"*pfadname*"

Mit diesem Parameter wird die Druckausgabe in eine Windows Datei umgeleitet. Es wird ab Dateianfang geschrieben, eine evtl. existierende Datei wird zuvor gelöscht.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pfadname Vollständiger Pfadname einer Windows Datei in die umgeleitet werden soll. Die Datei wird implizit eingerichtet.

Beispiele

```
DEVICE=8060,DRUA,"..\Drucker\drua0.txt";
DEVICE=8060:3,DRUA,"..\Drucker\drua0.txt";
```

Druckausgabe in **Windows Datei**

DEVICE = *ioadr*,DRSPOT,"*pfadname*",*timer*
DEVICE = *ioadr*,DRSPOTS,"*pfadname*",*timer*

Anmerkung

DRSPOT legt nicht DOSkompatible Dateinamen an.
(Namen > 8 Zeichen, Suffix > 3 Zeichen, z.B. 19980610123007.8028)
(Suffix .8028 = IO-Adr (Pseudo)Drucker in *mpar.sys*)
DRSPOTS legt DOSkompatible Dateinamen an.
(Namen <= 8 Zeichen, Suffix <= 3 Zeichen, z.B. 10123007.13)
(Suffix .13 = geräteindex = ZeilenNr. des Geräteeintrags in *mpar.sys*)

Mit diesem Parameter wird die Druckausgabe in eine Windows Datei umgeleitet. Bei der ersten Druckausgabe wird automatisch eine Datei mit dem Namen: TTSSMMss.iii (*tag, stunde, minute, sekunde.geräteindex* in *mpar.sys* /DRSPOTS/) oder
JJJJMMTTSSMMss.ioa (*jahr, monat, tag, stunde, minute, sekunde.io-adresse /DRSPOT/*) angelegt. Nach jeder Druckausgabe in die Datei wird ein Timer ‘timer’ neu aufgezogen. Läuft der Timer ab, wird die Datei geschlossen. Bei der nächsten Druckausgabe wird eine neue Datei angelegt.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pfadname Vollständiger Pfadname. Die Datei wird implizit eingerichtet.

timer Angabe in Sekunden, nach denen eine Datei nach der letzten Druckausgabe geschlossen wird. Anschließend wird eine neue Datei angelegt.

Beispiel

```
DEVICE=A358:3,DRSPOTS,"I:\Drucker\Nsu",30;
```


© IPKS GmbH 2007

Druckausgabe in **Windows Datei** (parametergesteuert)

DEVICE = *ioadr*,DRSPOTX,"*file*"

Mit DRSPOTX wird die Druckausgabe parametergesteuert in eine Windows Datei umgeleitet.

ioadr	IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).
file	(Pfad)Name der Parameterdatei.

Anmerkung

Beim Erstellen der Parameterdatei müssen folgende Regeln beachtet werden:
Kommentare beginnen mit einem Semikolon.
Die Parameternamen SUPPRESS, CHANGE, START, STOP können jeweils bis zu 255 mal verwendet werden, ein String kann bis zu 64 Zeichen lang sein, alle Zeichen sind erlaubt.
Die übrigen Parameternamen dürfen nur einmal verwendet werden, bei Mehrfachangaben gilt jeweils die letzte Definition.
In den Zeichenketten "string" können Sonderzeichen, wie unten beschrieben, als Hexamuster dargestellt werden:
Die beiden einem Backslash(****) nachfolgenden Zeichen werden als Hexamuster interpretiert. Das Zeichen **** selbst kann nur als \5d dargestellt werden.
Das Zeichen **"** kann nur als \22 dargestellt werden.
In einem Ausgabesatz wird nur ein START- oder STOPstring interpretiert.
Eine Änderung der Parameterdatei im laufenden Betrieb wird erst nach einem Neustart des Emulators wirksam oder durch das Auslösen der Funktion "Gerät rücksetzen" mit Hilfe des Programms TW.EXE. Vor der Funktion wird ggfs. eine gerade eröffnete Datei geschlossen.
Zeitgesteuerte (einmalig oder wiederkehrend) Aktivitäten müssen mit dem Windows Kommando **at** *zeit befehl* realisiert werden. Dazu muss der Zeitplandienst (SCHEDULE+) von Windows gestartet sein. (**net start** schedule)

Beispiel

at 23:00 netcopy.bat

Die dynamischen Funktionen werden in der Reihenfolge:
1. SUPPRESS: alle zu löschenden Zeichen werden entfernt
2. CHANGE: alle nach dem Löschen noch vorzunehmenden Ersetzungen werden ausgeführt
3. START: es wird geprüft, ob der Datensatz nach den Löschvorgängen und nach den Ersetzungen eine "Startzeichenfolge" enthält
4. STOP: es wird geprüft, ob der Datensatz nach den Löschvorgängen und nach den Ersetzungen eine "Stopzeichenfolge" enthält

bearbeitet.

Parameter:

PATH [*drive*:[*path*]]	; Pfadname unter dem die zu erzeugenden Dateien abgelegt werden. (Default: M2000 Startverz.)
SPOTTIME *sec*	; Einstellen der Überwachungszeit (default: 0, keine Überwachungszeit).
DOSNAMES	; Verwenden von DOS-kompatiblen Dateinamen,
NTNAMES	; ... nicht DOS-kompatiblen Dateinamen (default).
LFDNAMES	; Bilden der Dateinamen als ' laufende Nummer ' .
STARTNAMES *n*	; *n* Zeichen, die START "*string*" folgen, bilden den Anfang des Dateinamen. (z...z_JJJJMMTThhmmss.ioadr)
EXTENSION x....x	; Angabe einer frei wählbare (Datei)Extension.
SHARE	; ExklusivBelegStatus der AusgabeDateien wird aufgehoben.
SUPPRESS "*string*"	; Entfernen von Zeichen(folgen) vor der Ausgabe
CHANGE "*string1*" "*string2*"	; Ersetzen von Zeichen(folgen) vor der Ausgabe, *string1*/*string2* können verschieden lang sein.
START "*string*"	; Definition einer "Startzeichenfolge"
STOP "*string*"	; Definition einer "Stopzeichenfolge"
WINDOW "*name*"	; Ausgaben zusätzlich in Fenster *name*
WINDOWT "*name*"	; `dito`, zusätzlich mit vorangestelltem Zeitstempel

COMMAND "[*drive*:[*path*]] *file*" ; Definition eines Kommandos;

Das Kommando wird nach dem Schließen einer Zieldatei abgesetzt. Es können 5 Parameter mitgegeben werden:

der 1. Parameter bezeichnet die gerade geschlossene Datei,
der 2. Parameter enthält Kennungen wie,
Datei geschlossen nach Ereignis:
"to" *Zeitspanne*
"aa" *Startzeichenfolge*
"zz" *Stopzeichenfolge*
"bo" *ORG-Boot* oder *Drucker rücksetzen*
"nt" nach Beenden der Emulation

der 3. Parameter gibt nur das Verzeichnis an, in dem die Datei abgelegt ist
der 4. Parameter gibt nur den Namen der gerade geschlossenen Datei an
der 5. Parameter gibt nur die Erweiterung der gerade geschlossenen Datei an

Beispiel

```
DEVICE=A358:3,DRSPOTX,"\sicomp\spotx.par";
```

Die Parameter-Datei *spotx.par* könnte z.B. so aussehen:

```
PATH e:\temp                                                    ; ( 1 )
DOSNAMES                                                        ; ( 2 )
EXTENSION DRUA1                                                 ; ( 3 )
SPOTTIME 20                                                     ; ( 4 )
SUPPRESS "\0D\0A\03"                                            ; ( 5 )
CHANGE "\0D\0A" "\0A"                                           ; ( 6 )
START "//JOB:"                                                  ; ( 7 )
STOP  "//JEND"                                                  ; ( 8 )
STARTNAMES 7                                                    ; ( 9 )
SHARE                                                           ; ( 10 )
COMMAND "e:\temp\drspotx.bat" "drfile1" "to" "\temp" "drfile1" "txt"   ; ( 11 )
```

(1) Die Druckdateien sollen im Verzeichnis *e:\temp* abgelegt werden.
(2) Es sollen DOS-kompatible Dateinamen verwendet werden.
(3) Die frei gewählte (Datei)Erweiterung ist .DRUA1 statt *.ioadr .*
(4) Die Druckdatei wird geschlossen, wenn 20 sec keine Ausgabe erfolgt.
(5) Die Zeichenfolge CR-LF-ETX soll unterdrückt werden.
(6) Die Zeichenfolge CR-LF soll durch LF ersetzt werden.
(7) Die aktuelle Druckdatei wird geschlossen, wenn die Zeichenfolge "//JOB:" erkannt wird; eine neue Datei wird angelegt und "//JOB:" wird in die Neue übernommen.
(8) Die aktuelle Druckdatei wird geschlossen, wenn die Zeichenfolge "//JEND" erkannt wird und der String "//JEND" wird an das Ende der zu schließenden Datei geschrieben. Anschließend wird eine neue Druckdatei eröffnet, in die weitere Daten bis zum (jeweils gewünschten) Ausgabeende geschrieben werden.
(9) Die ersten 7 Zeichen, die "//JOB:" folgen (z.B. JOBCOPY), bilden den Anfang des neuen Dateinamens. z.B. *JOBCOPY_20020812135801.0202*
(10) Der (Datei)Status *geteilte Belegung* wird aktiviert, die Datei kann während der Ausgabe betrachtet werden.
(11) Die Druckdatei *drfile1.txt* wird wegen Kennung *to* nach Ablauf von Zeitspanne (SPOTTIME) geschlossen. Nach dem Schließen der Datei soll der Batch *drspotx.bat* aktiviert werden.

Anmerkung Bei SPOTTIME 0 (default) bleibt die zuletzt angelegte Datei solange offen und 'aufnahmefähig', bis andere Ereignisse (START ..., STOP ..., etc.) ein Schließen der Datei veranlassen.

 © IPKS GmbH 2007

Druckerbetrieb (seriell) mit **DR202**

DEVICE = *ioadr*,DR202,"*pfadname*"[,*flags*[,*timer*[,*verz1*]]]

Dieser Parameter ermöglicht den Betrieb eines an der seriellen Schnittstelle angeschlossenen Druckers DR202.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pfadname Pfadname der entsprechenden seriellen Schnittstelle unter Windows. Der *pfadname* wird in der Form "COMx" bzw. "\\.\COMxx" notiert.

Für DR202 kann die serielle Schnittstelle auch auf einem entfernten Rechner sein. Der *pfadname* wird in der Form “REM:x“ angegeben. Der Parameter x steht für die Portnummer des ersten (von zwei) TCP/IP - Sockets, den der Geräteprozess auf dem M2000 Rechner einrichtet (siehe Kapitel: Schnittstellenkonzentrator).

flags Parametrierung (Hexa)
= 1 unklar Erkennung bei TTY Schnittstelle
(Brücke zwischen RxD und CTS erforderlich)
= 2 DSR Abfrage vor Ausgabe auf V.24 Schnittstelle
= 4 ready/busy

timer Wert in Sek. für WRITE Überwachung (Default = 20Sec)

verz1 Verzögerungszeit in ms vor dem Senden von Daten (falls das angeschlossene Gerät zu langsam ist)

Anmerkung Bei einer Druckerparametrierung über das ORG (/PARAM..) werden von dem M2000 Treiberprozess die Byte 0,1,2,3,4 ausgewertet.

Beispiele Anschluss eines Druckers an die lokale COM Schnittstelle:

```
DEVICE=8060,DR202,"COM7";
DEVICE=8060:1,DR202,"COM7";
```

Anschluss an die COM-Schnittstelle eines entfernten Windows Systems:

```
DEVICE=8060,DR202,"REM:23000";
```

© IPKS GmbH 2007

Druckerbetrieb (seriell)
alle Drucker

DEVICE = *ioadr*,DRCOM,"*pfadname*"[,*typflags*[,*time1*[,*time2*]]]

Dieser Parameter ermöglicht den Betrieb aller an der seriellen Schnittstelle angeschlossenen Drucker.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pfadname Pfadname der entsprechenden seriellen Schnittstelle unter Windows. Der *pfadname* wird in der Form "COMx" bzw. "\\.\COMxx" notiert.

Für DRCOM kann die serielle Schnittstelle auch auf einem entfernten Rechner sein. Der *pfadname* wird in der Form “REM:x“ angegeben. Der Parameter x steht für die Portnummer des ersten (von zwei) TCP/IP - Sockets, den der Geräteprozess auf dem M2000 Rechner einrichtet (siehe Kapitel: Schnittstellenkonzentrator)

typflags Parametrierung (Hexa)
Hier kann entweder:
der Druckertyp in der Form ‘’typ’’ ,
der String ‘’CRLF’’, oder
die Flags:
= 1 Break-Erkennung bei TTY
= 2 Break-Erkennung bei V.24
= 4 ready/busy
= 8 seriellen Port nicht parametrieren, das MODE-Kommando ist gültig
= 10 wie Parameter(string) CRLF
= 20 SICOMP-Parameter (/PARAM) ignorieren

time1 Wert in Sek. für WRITE Überwachung (Default = 20Sec)

time2 Verzögerungszeit in ms vor dem Senden von Daten (falls das angeschlossene Gerät zu langsam ist)

Anmerkung Bei einer Druckerparametrierung über das ORG (/PARAM ..) werden vom M2000 Treiberprozess die Byte 0, 1, 2, 3, 4 ausgewertet.

Beispiele Anschluss eines Druckers an die lokale COM Schnittstelle

```
DEVICE=8060,DRCOM,"\\.\COM7",8,20,100;
DEVICE=8060:1,DRCOM,"COM7";
```

Anschluss eines Druckers an die COM-Schnittstelle eines entfernten Windows Systems

```
DEVICE=8060,DRCOM,"REM:23000";
```

 © IPKS GmbH 2007

Druckausgabe in ein **Windows Fenster**

DEVICE = *ioadr*,DRWIN,"*fenster*"[,"*datei1*","*datei2*","size"]
Mit diesem Parameter wird die Druckausgabe in ein Windows Fenster umgeleitet und zusätzlich in Dateien geschrieben.

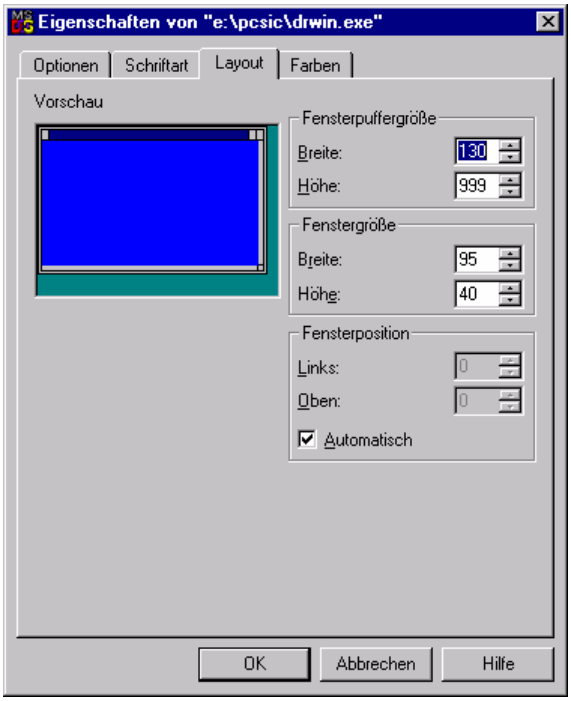
ioadr	IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)
fenster	Fenstername
datei1	erste Protokolldatei
datei2	zweite Protokolldatei
size	max. Dateigröße in Kilobyte

Beispiele

```
DEVICE=8060,DRWIN,"DRUA0","outfile1","outfile2","10";
DEVICE=A360:1,DRWIN,"DRUA0";
```

Der Parameter DRWIN besagt, dass die Druckausgaben in ein Windows Fenster umgeleitet werden. Nachdem Sie die Emulation gestartet haben, erscheint jetzt auf Ihrem Bildschirm ein weiteres Fenster mit dem Namen DRUA0. Dieses Fenster können Sie konfigurieren. Klicken Sie mit der rechten Maustaste die Kopfleiste des Fensters DRUA0 an. Wählen Sie die Funktion *Eigenschaften*. Über *Layout* ⇒ *Fensterpuffergröße - Fenstergröße* legen Sie fest, wieviel Druckerzeilen das Fenster puffern kann (beispielsweise 999). Das Windows Fenster arbeitet jetzt wie ein Umlaufpuffer mit einer Größe von 999 Zeilen.

Wenn Sie jetzt Teile der Druckerausgaben (beispielsweise Fehlermeldungen) auf einem am PC angeschlossenen Drucker ausgeben wollen, gehen Sie wie folgt vor:
Klicken Sie mit der rechten Maustaste die Kopfleiste an und wählen Sie die Funktionen *Bearbeiten* ⇒ *Markieren*. Markieren Sie im Fenster die Zeilen, die Sie drucken wollen. Übernehmen Sie den markierten Bereich in die Zwischenablage (CTRL und C). Rufen Sie ein Textprogramm auf wie beispielsweise WRITE. Übernehmen Sie dort den markierten Bereich in ein Dokument (CTRL und V) und drucken Sie das Dokument aus.



(Zusätzliche) Ausgabe in Protokolldateien:
Ist eine zusätzliche Ausgabe in Protokolldateien gewünscht, dann müssen die Parameter *datei1*, *datei2* und *size* angegeben werden.
(siehe Beispiel 1)
Zuerst erfolgt die Ausgabe in *datei1*. Hat diese die maximale Größe (*size**1024) überschritten, dann wird mit *datei2* weitergearbeitet. Ist diese auch voll, wird *datei1* gelöscht und neu beschrieben.
Es wird jeweils eine komplette Druckerzeile in die Datei geschrieben und erst danach auf die Größe abgefragt.
Beim Beenden der Emulation wird die letzte Einstellung in der Datei *fenster.par* (z.B. *DRUA0.par*) abgelegt.
Hat sich nach einem Neustart nichts an den Parametern geändert, dann werden die Ausgaben an die zuletzt geschriebene Protokolldatei angehängt und wie oben beschrieben fortgefahren.
Wurden Parameter geändert oder treten Probleme beim Öffnen der zuletzt geschriebenen Protokolldatei auf, dann wird mit der Ausgabe in die Datei *datei1* neu begonnen.
Beim Beenden wird eine Meldungszeile (--Ende DRWIN--) an das Ende der aktuellen Protokolldatei geschrieben.

Druckausgabe mit Datum/Uhrzeit in ein **Windows Fenster**

DEVICE = *ioadr*,DRWINT,"*fenster*"

DRWINT entspricht dem Parameter DRWIN; zusätzlich wird in jede Zeile der Druckausgabe am Zeilenanfang Datum und Uhrzeit im Format: `JJJJ-MM-TT hh:mm:ss.ms` geschrieben. Ideal für Meldedrucker.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

fenster Fenstername

Beispiele

```
DEVICE=8060,DRWINT,"DRUA";
```

Beispiel DRWINT

```
2002-08-1311:58:02.701 CDLI TO DRUA
2002-08-13 11:58:02.701 TESTCD 28 CA CBUET 65271 42
```

Hinweis Werden statt `DRWIN/DRWINT` die Parameter `DRWINMIN/DRWINTMIN` eingetragen, dann werden die DruckerFenster – minimiert – aktiviert.

© IPKS GmbH 2007

4.2.4 Kopplungen

Rechnerkopplung **DU02** **DEVICE**=*ioadr*,DU02,"SOCKET","*host*",*port*,*type*,0,*size*,[*t1*[,*t2*[,*t3*]]]

Gateway-Funktion

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU02 (Lichtwellenleiter-Kopplung) aktiviert.
Damit lassen sich zwei Emulator-Systeme verbinden.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

SOCKET für TCP/IP-Übertragung über Win-Socket

host hostname oder Leerstring (siehe 6.2.2 DU05 auf TCP/IP)

port Portnummer (ab 10 000...) (siehe 6.2.2 DU05 auf TCP/IP)

type
B: balanced
UP: unbalanced primary
US: unbalanced secondary

size Maximale Größe der ausgetauschten Anwenderdatensätze

t1 Wartezeit nach Verbindungsaufbau-Fehler in Millisekunden (Default: 10000 ms)

t2 Max. Wartezeit für Eingabebefehl vom ORG (Default: 5000 ms)

t3 Max. Wartezeit für Quittung vom Partnersystem (Default: 10000 ms)

Beispiel

```
DEVICE=8300,DU02,"SOCKET","WINNT1",22220,US,0,1024,30,5,0;
```
aktive Seite beim Verbindungsaufbau, Hostname eingetragen

```
DEVICE=8300,DU02,"SOCKET","",22221,US,0,1024,30,5,0;
```
passive Seite beim Verbindungsaufbau, Leerstring

Rechnerkopplung **DU03**

Für die Generierung einer DU03-Baugruppe waren im ORG-M vier Gerätenamen(Teilgeräte) zwingend (siehe unten). Über jedes Gerät werden unterschiedliche Kanäle angesprochen, hinter denen sich wiederum verschiedene Funktionalitäten verbergen. (z.B. *ISO-Leitung*, *SNVS-Leitung*)

Beispiel für die Generierung einer einzigen DU03 unter ORG-M:

```
/G:9400,KHDK0011=DU003,ST;
/G:9402,KHDK0012=DU003,ST;
/G:9404,KHDK0013=DU003,ST;
/G:9406,KHDK0014=DU003,ST;
```

Details sind für den Anwender der Emulation unwichtig, da immer alle generierten Teilgeräte emuliert werden.

DEVICE = *ioadr:*255,DU03,"NDIS",0; (WinNT)
DEVICE = *ioadr:*255,DU03,"NDIS2000",0; (Win2000)

ioadr	IO-Adresse des Gerätes gemäß ORG-Generierung(4stellige Hexazahl)
:255	fester Wert zum Maskieren der TeilgeräteAdresse
NDIS	die DU03-Emulation läuft unter WinNT
NDIS2000	die DU03-Emulation läuft unter Win2000

Die DU03-Emulation bildet die Original-Funktionalität –Datentransfer- vollständig ab. Nicht abgebildet sind Test- und Wartungsaufträge, die aber aus Kompatibilitätsgründen anzeigenfrei durchlaufen und abgeschlossen werden.
Der Betrieb einer DU03-Emulation in einem hochbelasteten Netz kann zu einem Performance-Verlust der GesamtEmulation führen. In dem Fall wäre der Einsatz von Switches, Bridges usw. notwendig, auch der eines Mehrprozessor-PC wäre (Performance)fördernd.
Die Rechnerkopplung DU03 wird ohne spezielle Hardware emuliert, es wird lediglich eine handelsübliche Netzwerkkarte benötigt. Daraus resultiert, dass eine emulierte DU03 auch mit 100 Mbit/s arbeiten kann.
Auf dem Emulator-PC muss ein Paket- oder NDIS-Treiber installiert werden, der von IPKS derzeit für WindowsNT und Windows2000 geliefert werden kann.
Mit Hilfe dieses Treibers kann die DU03-Emulation über die zugehörige Netzwerkkarte jedes empfangene Datenpaket lesen, auf Relevanz prüfen und verarbeiten, sowie Datenpakete versenden.

Benötigt werden für den Betrieb in WinNT/Win2000:

- die PacketDriver Software für den alten NDIS-Treiber(WinNT) oder
- die Software für den neuen Windows2000 PacketDriver
- installiert wird nach Angaben im mitgelieferten File *packet.htm*
- in der Parameterdatei *mpar.sys* wird zwischen NDIS(alt) und NDIS(neu) unterschieden

Beispiele

```
device = 9400:255,DU03,"NDIS",0;
device = 9400:255,DU03,"NDIS2000",0;
```

Als IO-Adresse*(ioadr)* ist immer die mit "00" (device=9400:255) endende Adresse anzugeben.

© IPKS GmbH 2007

Rechnerkopplung **DU04** über DF32/42

DEVICE = *ioadr*,DU04,"*pfadname*",*dfadr*[,*buflen*]

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU04 über eine Schnittstelle der DF32 bzw. DF42 definiert. Die DF32 bzw. DF42 muss mit dem Protokolltreiber COMDRV (3964R) installiert sein.
In diesem Fall wickelt die DF32/42 das 3964R Protokoll ab. Dementsprechend gilt die Beschreibung und Parametrierung dieser Baugruppe.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pfadname Pfadname des entsprechenden Windows-Gerätes.
Der Betrieb ist nur über eine DF32/42 möglich. Die Schnittstelle ist mit den Softwarehilfsmitteln von Comsoft (s. Handbuch DFxx) mit der entsprechenden Boardsoftware (Treiber COMDRV) zu laden und zu parametrieren. Die Einstellung der Schnittstelle wird von der Emulation nicht verändert.

dfadr Adressierung der Board-Schnittstelle in der Form:

bs
- *b* Boardnummer 0..4
- *s* Schnittstellennummer im Board 0..7

buflen Empfangspufferlänge für den DF32/42 Treiber in Bytes. Diese Länge muss mit dem bei der Protokolltreiberinstallation der DF32/42 angegebenen Wert übereinstimmen.
Defaultwert ist 1024.

Beispiel
```
DEVICE=8060,DU04,"\\.DFXX0",13;
```

Rechnerkopplung **DU04** über eine beliebige serielle Schnittstelle

DEVICE = *ioadr*,DU04,"*pfadname*",*par*
DEVICE = *ioadr*,DU04S,"*pfadname*",*par*

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU04 über eine beliebige serielle Schnittstelle definiert. Diese serielle Schnittstelle muss als COMxx unter Windows ansprechbar sein. Bei Problemen mit zu kleinen Wartezeiten bei der Rechnerkopplung **DU04** steht ab Emulatorversion V3.06 K025 ein zusätzliches "Gerät" **DU04S** zur Verfügung. Das "S" steht für "slow", d. h., der Emulator vergrößert die intern verwendeten Quittungszeiten.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pfadname Pfadname der entsprechenden Windows-Geräteschnittstelle, d.h. "COMx" bzw. "\\.\COMxx". Die Schnittstelle muss bereits mit dem Windows Kommando *mode* vorparametriert sein (Grundeinstellung). Eine Überparametrierung durch das ORG-M wird weitgehend durchgeführt.

par Protokollparameter in der Form:

wxpb
- w 0 = 4 KByte interner Puffer (default)
  1 = 64 KByte interner Puffer
- x 0 = keine BREAK Erkennung
  1 = TTY BREAK Erkennung
- p 0 = Priorität niedrig
  1 = Priorität hoch
- b 0 = ohne BCC
  1 = mit BCC

Beispiel
```
DEVICE=8060,DU04,"COM5",1011;
DEVICE=8060,DU04S,"\\.\COM15",11;
```

Rechnerkopplung **DU04** über TCP/IP

DEVICE = *ioadr*,DU04,"SOCKET",port-own,
host-remote:port-remote;

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU04 über TCP/IP und LAN oder RAS definiert. In Windows muss ein LAN / RAS Netzwerk-Adapter und TCP/IP installiert sein (siehe auch Kopplung DU05).

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

port-own Portnummer/Socketnummer im eigenen Rechner (z.B.: 22223).

host-remote Host Name oder Host Adresse im entfernten Rechner (z.B.: winnt1 oder 220.250.1.122).

port-remote Portnummer/Socketnummer im entfernten Rechner (z.B.: 22225).

Beispiele

```
DEVICE=8060,DU04,"SOCKET",22223,winnt7:22225;
DEVICE=8060,DU04,"SOCKET",22223,220.250.1.155:22225;
```

Rechnerkopplung **DU05** über DF42 Parametrierung über Emulator

DEVICE = *ioadr*,DU05, "\\.\DFXX0",*frame*,*parfile*

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU05 über eine Schnittstelle der DF42 definiert. Die DF42 muss mit einem Protokolltreiber MSV2 installiert sein. In diesem Fall wickelt die DF42 das MSV2 Protokoll ab. Dementsprechend gilt die Beschreibung und Parametrierung dieser Baugruppe.
Alternativ ist es möglich, die DU05 Kopplung auf eine TCP/IP Kopplung über die Win-Socket Schnittstelle umzulenken. (siehe Beispiel nächste Seite)
Neben diesem Geräteeintrag in der Konfigurationsdatei *mpar.sys* erfordert die DU05-Emulation eine Parameterdatei *parfile*, in der weitere Betriebsarten-Einstellungen definiert werden. Siehe dazu auch die Beschreibung in Kapitel 6.2.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

frame Maximale FRAME-Länge

parfile Bei DFXX0 Pfad- und Dateiname einer Parameterdatei gemäß 6.2

Beispiele

```
DEVICE=8300,DU05,"\\.\DFXX0",512,c:\sicomp\du05.par;
```

Rechnerkopplung **DU05** über DF42 Parametrierung über DFTEST

DEVICE = *ioadr*,DU05, "\\.\DFXX0",board,unit,frame,options

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU05 über eine Schnittstelle der DF42 definiert. Die DF42 muss mit einem Protokolltreiber MSV2 installiert sein. In diesem Fall wickelt die DF42 das MSV2 Protokoll ab. Dementsprechend gilt die Beschreibung und Parametrierung dieser Baugruppe.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung

board Boardnummer der Comsoft-Baugruppe

unit Schnittstellennummer der Comsoft-Baugruppe

frame Maximale FRAME-Länge

options z. Zt. immer 0

Beispiele

```
DEVICE=8300,DU05,"\\.\DFXX0",0,0,1024,0;
```

 © IPKS GmbH 2007

Rechnerkopplung **DU05** Gateway-Funktion

DEVICE=*ioadr,*DU05,*"*SOCKET","*host",port*[*/t0*],*size,*
[*t1*[,*t2*[,*t3*]]]

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU05 auf eine TCP/IP Kopplung über die Win-Socket Schnittstelle umgelenkt. In diesem Fall wird eine standardmäßige LAN-Kopplung verwendet.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

"SOCKET" für TCP/IP-Übertragung über Win-Socket

host hostname oder Leerstring (siehe 6.2.2 DU05 auf TCP/IP)

port Portnummer (ab 10 000...) (siehe 6.2.2 DU05 auf TCP/IP)

t0 Verbindungs-Überwachungszeit in Sekunden (Default: keine Überw.)

size Maximale Größe der ausgetauschten Anwenderdatensätze

t1 Wartezeit nach Verbindungsaufbau-Fehler in Millisekunden (Default: 10000 ms)

t2 Max. Wartezeit für Eingabebefehl vom ORG (Default: 5000 ms)

t3 Max. Wartezeit für Quittung vom Partnersystem (Default: 10000 ms)

Beispiel

```
DEVICE=8300,DU05,"SOCKET","HOST1",22220,512,20000,10000,30000;
```
aktive Seite beim Verbindungsaufbau, Hostname eingetragen
```
DEVICE=8300,DU05,"SOCKET","",22221,512,20000,10000,30000;
```
passive Seite beim Verbindungsaufbau, Leerstring

Anmerkung Sind im Rechner zwei (oder mehr) Netzwerkkarten installiert, kann auch gezielt über die IP-Adresse (oder den Alias-Namen) der jeweiligen Karte eine Verbindung aufgebaut werden. Der Alias-Name muss in der Datei *hosts* in **systemdir***system32**drivers**etc*\... eingetragen sein. Die zugehörige Parameterzeile ist folgendermaßen aufgebaut:

DEVICE=*ioadr,*DU05, *"*SOCKET","[partner]-*ipadr*",*port,size*
DEVICE=*ioadr,*DU05, "SOCKET","[partner]-*aliasname*",*port,size*

Beispiel

```
DEVICE=B400,DU05,"SOCKET","partner-11.12.13.1",20000,2048;
```

```
DEVICE=B400,DU05,"SOCKET","partner-LAN1",20000,2048;
```
Verbindung wird über Netzwerkkarte 1 aufgebaut.

```
DEVICE=B400,DU05,"SOCKET","partner-11.12.13.2",20000,2048;
```

```
DEVICE=B400,DU05,"SOCKET","-11.12.13.2",20000,2048;
```

```
DEVICE=B400,DU05,"SOCKET","partner-LAN2",20000,2048;
```
Verbindung wird über Netzwerkkarte 2 aufgebaut.

© IPKS GmbH 2007

Rechnerkopplung **DU06** über DF32/42

DEVICE = *ioadr,*DU06, "*driver*",*kenn1,kenn2,type,*0,*frame,t1,t2,t3,parfile*

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU06 über eine Schnittstelle der DF32 bzw. DF42 definiert. Die DF32 bzw. DF42 muss mit einem Protokolltreiber HDLC UNC-2 oder HDLC BAC 2,8 installiert sein. In diesem Fall wickelt die DF32/42 das HDLC-Protokoll ab. Dementsprechend gilt die Beschreibung und Parametrierung dieser Baugruppe.
Neben diesem Geräteeintrag in der Konfigurationsdatei *mpar.sys* erfordert die DU06-Emulation eine Parameterdatei *parfile*, in der weitere Betriebsarten-Einstellungen definiert werden. Siehe dazu auch die Beschreibung in Kapitel 6.3 DU06.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

driver "\\.\DFXX0" für COMSOFT DFxx

kenn1 Baugruppennummer DFxx

kenn2 Schnittstellennummer der Baugruppe DFxx

type
- B: balanced
- UP: unbalanced primary
- US: unbalanced secondary

frame Maximale I-FRAME-Länge

t1
- 0: Keine Eingabeüberwachung
- 1: Eingabeüberwachung 1 Sekunde (Meldung)
- 5: Eingabeüberwachung 5 Sekunden (Meldung)
- 24: Eingabeüberwachung 24 Sekunden (Meldung)
- 101: Eingabeüberwachung 1 Sekunde (Meldung/Verbindung trennen)
- 105: Eingabeüberwachung 5 Sekunden (Meldung/Verbindung trennen)
- 124: Eingabeüberwachung 24 Sekunden (Meldung/Verbindung trennen)

t2
- 0: Wählverbindungen in Übertragungspausen nicht trennen
- 1: Wählverbindungen in Übertragungspausen trennen

t3 Quittungsüberwachungszeit in Millisekunden

parfile Pfad- und Dateiname einer Parameterdatei gemäß 5.3

Hinweise Derzeit sind die Timer t1, t2 und t3 mit 0 zu besetzen !

Beispiel

```
DEVICE=8300,DU06,"\\.\DFXX0",0,0,US,0,0,0,0,
g:\sicomp\du06us.par;
```

 © IPKS GmbH 2007

Rechnerkopplung **DU06** **DEVICE**=*ioadr*,DU06,"SOCKET","*host*",*port*,*type*,0,*size*,[*t1*[,*t2*[,*t3*]]]

Gateway-Funktion

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DU06 auf eine TCP/IP Kopplung über die Win-Socket Schnittstelle umgelenkt. In diesem Fall wird eine standardmäßige LAN-Kopplung verwendet.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

SOCKET für TCP/IP-Übertragung über Win-Socket

host hostname oder Leerstring (siehe 6.4.4 DU06 auf TCP/IP)

port Portnummer (ab 10 000...) (siehe 6.4.4 DU06 auf TCP/IP)

type
B: balanced
UP: unbalanced primary
US: unbalanced secondary

size Maximale Größe der ausgetauschten Anwenderdatensätze

t1 Wartezeit nach Verbindungsaufbau-Fehler in Millisekunden (Default: 10000 ms)

t2 Max. Wartezeit für Eingabebefehl vom ORG (Default: 5000 ms)

t3 Max. Wartezeit für Quittung vom Partnersystem (Default: 10000 ms)

Beispiel

```
DEVICE=8300,DU06,"SOCKET","WINNT1",22220,B,0,1024,30,5,0;
```
aktive Seite beim Verbindungsaufbau, Hostname eingetragen
```
DEVICE=8300,DU06,"SOCKET","",22221,B,1024,30,5,0;
```
passive Seite beim Verbindungsaufbau, Leerstring

Anmerkung

Sind im Rechner zwei (oder mehr) Netzwerkkarten installiert, kann auch gezielt über die IP-Adresse (oder den Alias-Namen) der jeweiligen Karte eine Verbindung aufgebaut werden. Der Alias-Name muss in der Datei *hosts* in **systemdir***system32**drivers**etc*\... eingetragen sein. Die zugehörige Parameterzeile ist folgendermaßen aufgebaut:

DEVICE=*ioadr*,DU06,"SOCKET","[partner]-*ipadr*",*port*,*type*,*0*,*size*
DEVICE=*ioadr*,DU06,"SOCKET","[partner]-*aliasname*",*port*,*type*,*0*,*size*

Beispiel

```
DEVICE=B400,DU06,"SOCKET","partner-11.12.13.1",20000,B,0,1024;
```

```
DEVICE=B400,DU06,"SOCKET","partner-LAN1",20001,B,0,1024;
```
Verbindung wird über Netzwerkkarte 1 aufgebaut.

```
DEVICE=B400,DU06,"SOCKET","partner-11.12.13.2",20000,B,0,1024;
```

```
DEVICE=B400,DU06,"SOCKET","-11.12.13.2",20000,B,0,1024;
```

```
DEVICE=B400,DU06,"SOCKET","partner-LAN2",20001,B,0,1024;
```
Verbindung wird über Netzwerkkarte 2 aufgebaut.

© IPKS GmbH 2007

Kommunikations-steuerung **KS100**

DEVICE = *ioadr*,KS100,*index*,*ptime*,*pmax*,*0*

Mit diesem Parameter wird die Emulation einer KS100 Kommunikation über SINEC H1 definiert. Dazu muss ein CP1413/CP1613 installiert und parametriert sein. KS100 läuft unter SIMATIC NET.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

index Laufende Nummer (1 oder 2) des CP1413/CP1613 gemäß Windows-Parametrierung

ptime Wartezeit in Millisekunden auf Ergebnisse synchron auszuführender CALL_INA Aufrufe
Default: 50 ms (bei ptime = 0)

pmax Maximale Zyklen beim Warten auf Ergebnisse synchron auszuführender CALL_INA-Aufrufe
Default: 100 Versuche (bei pmax = 0)

Beispiel

```
DEVICE=8060,KS100,1,0,0,0;
```

Kommunikations-steuerung **CS275**

DEVICE = *ioadr*,CS275,*0,1*

Mit diesem Parameter wird der Anschluss an das TELEPERM M Bussystem CS275 definiert. Hierzu muss eine NAT Baugruppe installiert und parametriert sein.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

Die weiteren Parameter sind fest vorgegeben.

Beispiel

```
DEVICE=8060,CS275,0,1;
```

Kommunikations-steuerung **CP1400**

DEVICE = *ioadr*,CP1400,*par1,par2*

Mit diesem Parameter (CP1400) wird die Emulation einer SINEC H1 Kopplung über die ISO Protokolle 1-7 definiert. Dazu muss ein CP1413/CP1613 installiert und parametriert sein (siehe auch Kapitel: Kommunikationssteuerung CP1400).

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

par1 CP1413/CP1613 Baugruppennummer (1 oder 2)

par2 0

Beispiel

```
DEVICE=8060,CP1400,1,0;
```

 © IPKS GmbH 2007

Kommunikations-prozessor **UCP-2**

DEVICE = *ioadr*,UCP-2,*prsize*,*blsize*,*opt*

Mit diesem Parameter wird die Emulation eines UCP-2, d.h. einer TCP/IP Kommunikation, definiert. Die Geräteemulation leitet die Kommunikationsverbindungen auf die Windows Socket-Schnittstelle um.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

prsize Mindestgröße des Geräteprozesses unter Windows in KB (z.B.: 2048). Über diesen Parameter können zur Verbesserung der Performance Pufferbereiche im Geräteprozess reserviert werden. Bei der Angabe 0 wird keine Mindestgröße eingestellt.

blsize Maximale Sendepuffergröße in KB beim Aufruf SENDIT (Summe der Längen aller Teilpuffer). Default: 4 KB (bei blsize = 0). Für alle anderen Aufrufe werden die vom SICOMP System übergebenen Längen verwendet.

opt

0 ohne Optionen

1 Sockets mit SO_KEEPALIVE und SO_REUSEADDR einrichten

2 Bearbeitungsthread mit gleicher Priorität wie den Hauptthread starten

4 Bei Socket-Auftrag RECEIVEIT wird an der Schnittstelle zu Windows-TCP/IP immer die Puffergröße **32768** statt der vom Anwenderprogramm vorgegebenen Pufferlänge verwendet. Die Datenübergabe an das Anwenderprogramm erfolgt in Stückelungen gemäß der vom Anwenderprogramm vorgegebenen Pufferlänge.
Damit lässt sich bei bestimmten Anwendungen, (FTP-Transfer sehr großer Dateien) ein Performance-Gewinn erzielen.

Hinweise

- Werte des Parameters *opt* werden hexadezimal interpretiert, die Bitmuster können mit ODER verknüpft werden.
- Die Aufrufe zum Firmware-Laden können unter der Emulation entfallen, ansonsten werden die von der Emulation empfangenen Firmwaredaten ignoriert.
- Bei Verwendung von FTP (SICOMP-Programme UXFTP und UXFTPD) darf im Windows-Netzwerk das FTP-Protokoll nicht unter den Standard-Portnummern aktiviert werden.
- Bei der Verwendung von Standard-Diensten muss dieser in der Datei *%systemroot%\drivers\etc\services* auskommentiert oder mit einer anderen als der Standard-Portnummer versehen werden, damit dieser Dienst nur unter ORG-M, nicht aber unter Windows aktiviert ist.

Beispiel: Bei der Verwendung von Remote-Login (SICOMP-Programm RDATR) muss in der Datei *%systemroot%\drivers\etc\services* die Zeile 'login 513/tcp' auskommentiert werden.

Beispiele

```
DEVICE=8070,UCP-2,0,0,0;
DEVICE=8070,UCP-2,2048,0,0;
```

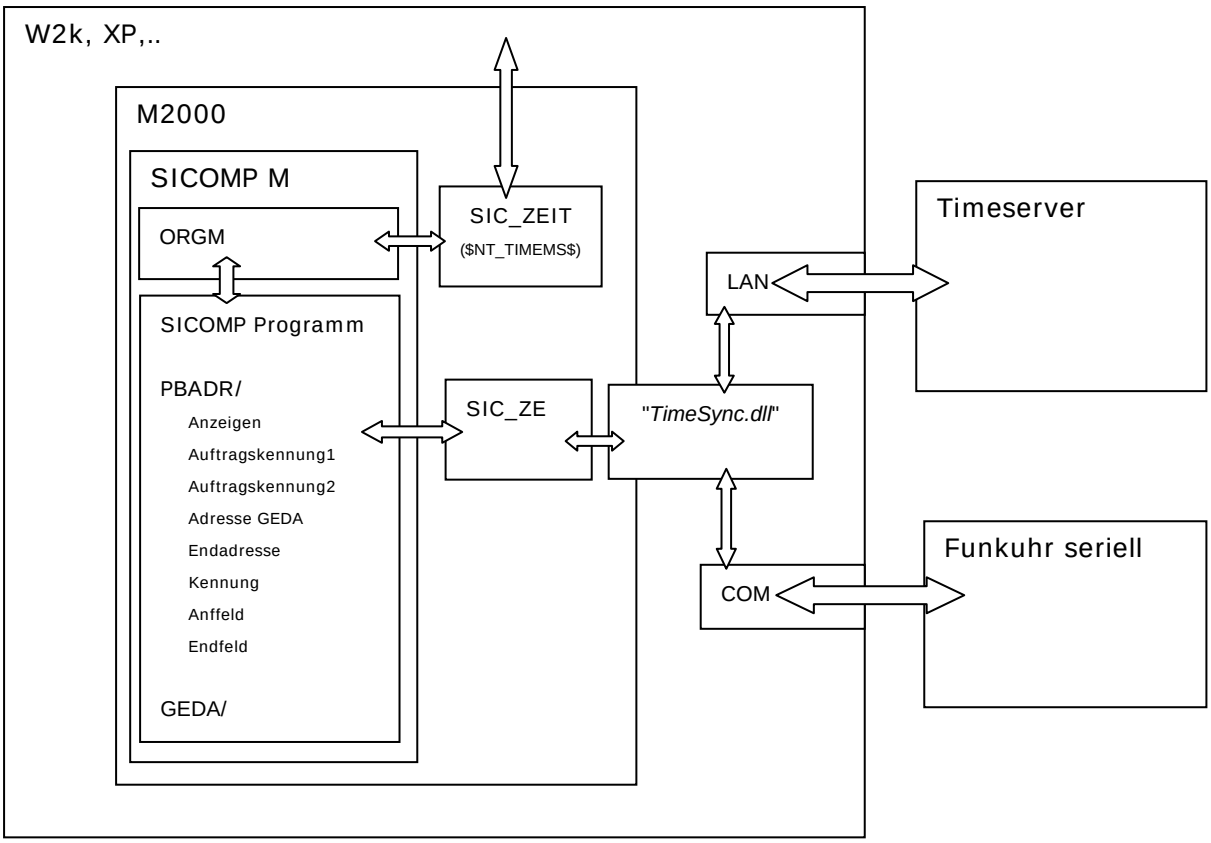
© IPKS GmbH 2007

4.2.5 Sonstige Geräte

4.2.5.1 ZeitSynchronisation

Zeit-Synchronisation

Der Parameter TIMESYNC ermöglicht das (Zeit)Synchronisieren des emulierten Systems mit Fremdsystemen - hier mit Windows – über Funkuhr oder TimeServer. Ein eigenständiger Emulatorprozess ermittelt über eine DLL (*TimeSync.dll*) auf Windows-Ebene die aktuelle Zeit und stellt diese einem SICOMP Programm zur Verfügung. Über den ORG-Aufruf $DATE werden SICOMP Zeit und die Sommer-/Winterzeit Kennung im ORG gesetzt, der Emulator aktualisiert dann die Windows Zeit mit der ORG Zeit.
Ausführliche Beschreibung der Zeitsynchronisation in einer gesonderten Dokumentation. (TimeSync.pdf in **CDroot**\Emulation\DOC auf der M2000 CD)



 © IPKS GmbH 2007

4.2.5.2 Zeitgeber

Promea-**Zeitgeber**

```
DEVICE = ioadr,ZIG1[,"pfadname"];
DEVICE = ioadr,ZIG2;
DEVICE = ioadr,ZIG3;
```

Die Parameter ZIG1/ZIG2/ZIG3 definieren einen Promea-Zeitgeber. Fehlt der Parameter, so wird die IO-Adresse 8010 angenommen.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pfadname Wird hier ein Pfadname angegeben, so merkt sich die Emulation beim Setzen der Uhrzeit unter ORG-M die Relativzeit zur Windows-Zeit in der Datei *pfadname*. Wird kein Pfadname angegeben, so geht die Relativzeit nach Beenden der Emulation verloren. Bei einem Neustart arbeitet die Emulation wieder mit der Windows-Zeit.
NT_TIME als Pfadname bewirkt, dass bei jedem Setzen des ORG Datums/der ORG Uhrzeit auch das Windows Datum/die Windows Uhrzeit gesetzt wird.
NT_TIMEMS als Pfadname bewirkt, dass zwei hintereinander folgende Abfragen keine Zeitinformation mit gleichem *ms* Wert bekommen (wg. Zeitstempel in DVS).
Wird SYS_TIME eingetragen, werden Datum/Uhrzeit unabhängig von Zeitzone und Sommer-/Winterzeit als absolute UTC Time (koordinierte Weltzeit, früher GMT) vom ORG interpretiert.

ZIG1 ist ein Zeitgeber, der mit der Genauigkeit [1)] der Windows Zeitauflösung arbeitet (10ms/15ms). Zeitabgleich je nach Zusatzparameter ("*pfadname*").

ZIG2 ist ein "millisekundengenauer" Zeitgeber beim Uhrzeit lesen. Parameter, Relativzeit usw., werden ignoriert.
ZIG2 arbeitet implizit mit NT_TIME. Die Genauigkeit von Kurzzeit- und Zykl.Wecker entspricht der Genauigkeit der Windows Zeitauflösung.

Problem *Abhängig von der PC Hardware (Prozessor, Taktgeber, etc.) kann die "Zeit weglaufen".*

ZIG3 entspricht ZIG1 mit (implizit) NT_TIME, aber:
ZIG3 arbeitet generell "millisekundengenau" [1)] durch Einstellung der Windows Zeitbasis auf eine MilliSekunde.

Beispiele

```
DEVICE=8010,ZIG1,"orgzeit";
DEVICE=8050,ZIG1,"$NT_TIME$";
DEVICE=8050,ZIG1,"$NT_TIMEMS$";
DEVICE=8050,ZIG1,"$SYS_TIME$";
DEVICE=8010,ZIG2;
DEVICE=8010,ZIG3;
```

Anmerkung Wenn das ORG Datum/Uhrzeit liest, liest der Emulator Datum/Uhrzeit von Windows. Wenn über das ORG Datum/Uhrzeit verändert werden, verändert der Emulator Datum/Uhrzeit von Windows nur dann, wenn als Pfadname NT_TIME bzw. NT_TIMEMS angegeben ist. Sollen ORG Datum/Uhrzeit und Windows Datum/Uhrzeit synchron laufen, muss der Pfadname NT_TIME bzw. NT_TIMEMS angegeben werden. Eine Uhrzeitverstellung, z.B. Sommer/Winterzeit, muss grundsätzlich vom ORG initiiert werden.
(serielle Funkuhr an PROMEA, manuelle Zeitverstellung oder über *TimeSync.dll*)
☐ Uhr automatisch auf Sommer/Winterzeit umstellen - muss grundsätzlich deaktiviert sein.

1) Die jeweilige Genauigkeit gilt für: Zyklischen- und Kurzzeit-Wecker, Uhrzeit lesen.

4.2.5.3 Profibus & Pseudo Geräte

Prozesselement **PE F7** (Profibus)

DEVICE = *ioadr*, ALEM, "PROFIBUS", *mode*, *polltime*

Dieser Parameter definiert eine PE F7 Geräteemulation über eine **Profibus**-Anschaltung.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl) , i.a. 2000H

mode Modus, hexadezimal, z. Zt. nur 0 sinnvoll

polltime Zykluszeit in Millisekunden, nach der alarmbildende (Eingabe-) Signalformer auf Veränderungen abgefragt werden.

Wichtig! Für die Rangierung der Prozesssignalformer erfordert die PE F7 Emulation zusätzliche Rangiereinträge

psf =.....

in der Konfigurationsdatei *mpar.sys*.
(siehe Kapitel: Emulation PE F7)

Pseudogerät für **PSD**

DEVICE = *ioadr*,FTNT,"*pipename*",*length*[,*vz*[,*br*[,*timeout*]]]

Mit dem Parameter FTNT wird ein Pseudogerät für PSD definiert (Pipe zu SICOMP Daten). Im SICOMP M System muss dafür ein Sichtgerät DSSK generiert sein. M2000 (*sic_ftnt.exe*) richtet dazu für jede Bearbeitungsrichtung eine *named pipe* unter Windows ein und arbeitet bezüglich dieser Pipe als **Server**. Die Pipe, über die Daten gelesen werden, wird mit der Endung **in** ergänzt, die Pipe, über die Daten geschrieben werden, wird mit der Endung **out** ergänzt.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

pipename Als Pipename sind bei Benutzung der *m2psapi.dll* nur die Namen "ftnt0" bis "ftnt9“ möglich (siehe auch: PSD Dokumentation). M2000 richtet für jedes Pseudogerät zwei Pipes ein, wobei der Pipename jeweils mit den Endungen **in** und **out** ergänzt wird. Damit steht für jede Richtung eine Pipe zur Verfügung. Wird ohne PSD-Aufrufe gearbeitet, dann gibt es keine Begrenzung bei der PipeNummern Vergabe.

length Länge der Pipes in KByte. Die Angabe 0 wird durch den Defaultwert von 4 KB ersetzt.

vz AusgabeVerzögerungsZeit in Millisekunden.

br Nicht angegeben oder ' 0 ' : bei Basisparametrierung werden die Pipes nicht geschlossen.
Angegeben und nicht ' 0 ' : bei Basisparametrierung werden die Pipes geschlossen.

timeout TimeOut bei "Überwachung Anruf Eingabe" in ms

Hinweis Wird `timeout` parametriert, müssen `vz` und `br` angegeben werden.

Für das (Pseudo)Gerät FTNTA gilt: es werden Anzeigen zurückgegeben, wenn ein Leseaufruf eintrifft und keine Daten vorhanden sind.

 © IPKS GmbH 2007

Pseudogerät für **PSD** **DEVICE** = ioadr,FTNTX,"*pipename*",*length*[,*vz*[,*br*[,*timeout*]]]

Mit diesem Parameter wird ein Pseudogerät für PSD definiert (Pipe zu SICOMP Daten). Im SICOMP M System muss dafür ein Sichtgerät DSSK generiert sein. M2000 (*sic_ftnt.exe*) greift hier auf eine vorhandene *named pipe* unter Windows zu und arbeitet bezüglich dieser Pipe als **Client**. Die Pipe, über die Daten gelesen werden, wird mit der Endung **out** ergänzt, die Pipe, über die Daten geschrieben werden, wird mit der Endung **in** ergänzt.

ioadr wie zuvor.

pipename wie zuvor, zusätzlich mit Angabe des entfernten Rechners. Es wird keine Pipe eingerichtet.

length wie zuvor.

vz wie zuvor.

br wie zuvor.

timeout TimeOut bei "Überwachung Anruf Eingabe" in ms

Wichtig! Standard TimeOut bei "Überwachung Anruf Eingabe" - default **10** sec, wird, wenn parametriert, mit Wert aus *timeout* überschrieben.

Hinweis Für das (Pseudo)Gerät FTNTAX gilt:
es werden Anzeigen zurückgegeben, wenn ein Leseaufruf eintrifft und keine Daten vorhanden sind.

mit **HardWare(Timer)** **DEVICE** = *ioadr,*FTNTHW,"*pipename*",*length*[,*vz*[,*br*[,*timeout*]]]
DEVICE = *ioadr,*FTNTXHW,"*pipename*",*length*[,*vz*[,*br*[,*timeout*]]]

alle Parameter wie oben, aber:

Standard TimeOut bei "Überwachung Anruf Eingabe" - default **120** sec, wird, wenn parametriert, mit Wert aus *timeout* überschrieben.

Der Zusatz HW im Geräteparameter steht für HardWare(ÜberwachungsTimer).

© IPKS GmbH 2007

Pseudogerät als **PipeServer/Client**

DEVICE = *ioadr*,FTNTD,*"pipename",length[,vz[,br[,timeout]]]*
DEVICE = *ioadr*,FTNTDX,*"pipename",length[,vz[,br[,timeout]]]*

FTNTD richtet ein Pipe-Paar ein und bearbeitet das jeweilige Pseudogerät als **Pipe-Server**.
FTNTDX stellt Verbindungen zu einem Pipe-Paar her und bearbeitet das jeweilige Pseudogerät als **Pipe-Client**.
Beide erwarten kein Längenwort vor den eigentlichen Anwenderdaten, sondern sie stellen das Längenwort beim Senden den Anwenderdaten voran und entfernen es beim Empfang - siehe Anmerkungen. Alle Parameter wie oben.

Beispiele

```
DEVICE=8058,FTNT,"ftnt0",0,700,1,1000 ;DSSK8 Pseudogerät für PSD;
DEVICE=A308:1,FTNT,"ftnt0",0,0,0,1000 ;DSSK9 Pseudogerät für PSD;
DEVICE=8028,FTNTX,"\\server1\pipe\ftnt0",0
                                      ;DSSK3 Pseudogerät für PSD;
DEVICE=8058,FTNTD,"ftntd0",0          ;DSSK8 Pseudogerät für PSD;
DEVICE=8028,FTNTDX,"\\server1\pipe\ftntd0",0;
                                      ;DSSK3 Pseudogerät für PSD;
```

Wichtig! Standard TimeOut bei "Überwachung Anruf Eingabe" - default **10** sec,
wird, wenn parametriert, mit Wert aus *timeout* überschrieben.

Hinweis Wird `timeout` parametriert, müssen `vz` und `br` angegeben werden.

Für das (Pseudo)Gerät FTNTDA bzw. FTNTDAX gilt:
es werden Anzeigen zurückgegeben, wenn ein Leseaufruf eintrifft und keine Daten vorhanden sind.

mit **HardWare(Timer)**

DEVICE=*ioadr,*FTNTDHW*,*"*pipename*",*length*[,*vz*[,*br*[,*timeout*]]]*
DEVICE=*ioadr,*FTNTDXHW*,*"*pipename*",*length*[,*vz*[,*br*[,*timeout*]]]*

Alle Parameter wie vorher, aber:
Standard TimeOut bei "Überwachung Anruf Eingabe" - default **120** sec,
wird, wenn parametriert, mit Wert aus *timeout* überschrieben.

Anmerkungen Die als Pseudogeräte verwendeten DSS-Anschlüsse sollten unter ORG-M mit automatischer Minusquittierung generiert werden. Änderungen im generierten System können mit TESTSM durchgeführt werden. Diesen *Eingriff ins ORG* sollte aber nur ein *Systemkenner* machen.

Über Remote - Pipes können zwei SICOMP Systeme unter M2000 direkt miteinander Daten austauschen. Auf Rechner 1 wird FTNT parametriert und auf Rechner 2 wird FTNTX parametriert. Es können so mit ''STAUAL Pseudogerät'' auf Rechner 1 Daten ausgegeben werden, die mit ''STEIAL Pseudogerät'' auf Rechner 2 eingelesen werden und umgekehrt. Die Übertragung über das Rechnernetz übernimmt die Windows Funktion Remote - Pipe.

Um bereits existierender SICOMP-Software die Möglichkeit zu geben, auch mit ''Pseudogeräten'' (statt mit realen Geräten) zu verkehren, wurde PSD durch zwei Pseudogerät-Parameter, FTNTD und FTNTDX, ergänzt. Die Konvention ''Längenwort'' (siehe nächste Seite) muss bei Einsatz dieser beiden Parameter nicht beachtet werden (siehe oben).

Anwender, die nicht mit dem Standardprogramm *PSDXXX* arbeiten, müssen bei der Programmierung sowohl unter Windows, als auch unter ORG-M, ORG-PV und ORG-HV eine Konvention beachten: den eigentlichen Daten muss ein Wort (SICOMP-Format) vorangestellt werden, das die Gesamtlänge der Daten in Worten beinhaltet (Längenwort). Diese Konvention hat sich aus der Telegrammschnittstelle zwischen *PSDXXX* und PSAPI entwickelt.

 © IPKS GmbH 2007

Pseudogerät für **WINCC-PSD**

DEVICE = *ioadr*,WINCC,"*pipename*",*length*[,*vz*[,*br*[,*timeout*]]]

Mit dem Parameter WINCC wird ein Pseudogerät für WINCC-PSD definiert. Es arbeitet im Prinzip wie FTNT, verwendet aber andere Zeitstufen. Im SICOMP M System muss dafür ein Sichtgerät DSSE/DSSA generiert sein. M2000 richtet dazu für jede Bearbeitungsrichtung eine *named pipe* unter Windows ein und arbeitet bezüglich dieser Pipe als **Server**. Die Pipe, über die Daten gelesen werden, wird mit der Endung **in** ergänzt, die Pipe, über die Daten geschrieben werden, wird mit der Endung **out** ergänzt.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

pipename Als Pipename sind beliebige Namen möglich, M2000 richtet für jedes Pseudogerät zwei Pipes ein, wobei der Pipename jeweils mit den Endungen **in** und **out** ergänzt wird. Damit steht für jede Richtung eine Pipe zur Verfügung.

length Länge der Pipes in KByte. Die Angabe 0 wird durch den Defaultwert von 4 KB ersetzt.

vz Ausgabeverzögerungszeit in Millisekunden.

br Nicht angegeben oder ' 0 ' : Arbeitsweise wie bisher.
Angegeben und nicht ' 0 ' : Bei Basisparametrierung werden die Pipes geschlossen.

timeout TimeOut bei "Überwachung Anruf Eingabe" in ms

Beispiel `DEVICE=8058,WINCC,"wincc0",0,100,,2000` ; Pseudogerät für PSD

Anmerkung Wenn von *PSDXXX* oder *WCCXXX* ein Schreibauftrag für den symbolischen CB-Namen "@@@@@@" oder die Objektnummer "9999" empfangen wird, dann wird mit den zugehörigen Parametern "Adresse" und "Wert" ein Anwenderunterprogramm *UPAWP* aufgerufen, das zu *PSDXXX* oder *WCCXXX* gebunden werden muss. Schreibaufträge müssen jeweils für ein einzelnes Wort abgesetzt werden, Leseaufträge sind unzulässig.

Für Anwender von WinCC und WinCC-PSD-Kanal-DLL bedeutet dies, dass man beispielsweise eine externe PSD-Variable des Typs 'A' (vorzeichenloser 16-Bit Wert) definieren kann, der der CB "@@@@@@" und die Adresse "0"(=V128) zugeordnet werden. Wenn man in diese Variable den Wert "100" schreibt (WinCC-Funktion "SetTagWord"), dann wird unter ORG-M oder ORG-PV das Unterprogramm *UPAWP* mit G6:=0 und G7:=100 aufgerufen. Damit können Anwender aus WinCC heraus beliebige (selbstprogrammierte) Funktionen unter dem emulierten ORG aktivieren. Als Kriterien dienen die V-Adresse in "Pseudo-CB" 9999 und der übergebene Wert.
PSD-Variablen, die dem CB "@@@@@@" zugeordnet werden, dürfen von WinCC nicht aktualisiert werden!
Die Standardelemente *PSDXXX* und *WCCXXX* enthalten ein Dummy-Modul. Dieses Dummy-Modul wird auf der virtuellen Platte PLSK.PSD als Element *PLSKx-PSD.PSDAWP* mitgeliefert und kann dem Anwender als Muster dienen.
Es gibt lediglich die übergebenen Werte zu G6 und G7 auf dem Standardmeldegerät aus.

© IPKS GmbH 2007

Beispiel Über WinCC-Buttons soll das ORG mit oder ohne Startliste gebootet werden können. Der PSD-Variablen "ORGBOOT" (Typ "vorzeichenloser 16-Bit-Wert") werden CB "@@@@@@" und Adresse "3" (=V130) zugeordnet. Ein Button setzt den Wert 1 (SetTagWord (ORGBOOT,1)), ein anderer Button den Wert 2 (SetTagWord (ORGBOOT,2)). In *UPAWP* wird zunächst geprüft, ob es sich um die Funktion "ORG booten" handelt, indem G6 mit 3 verglichen wird. Anschließend wird über G7 (1 oder 2) festgestellt, ob mit oder ohne Startliste gebootet werden soll, danach wird die Funktion entsprechend ausgeführt ($RUFORG"...).

Hinweis Wenn die alten Binderkarten mit den neuen Grundsprache-Elementen verwendet werden, kommt es beim Laden von *PSDXXX* und *WCCXXX* zur Fehlermeldung: "/NDEF:UPAWP". Diese Meldung kann ignoriert werden, solange die neue Funktion nicht benutzt werden soll.

Dummy-Modul

```
/#PSDAWPXX                          170800/PJG
************************************************************************
***         PSDXXX:   ANWENDERFUNKTION
************************************************************************
          'PKEN' ,,,,0
          'DEEX' UPAWP
          'NDEX' STAMELD

          'NAME' PSDAWP
          'SATZ' PSDAWP

'VA' TEXTA/
' Y'         = 'Z=PSDAWP: DUMMY: #### ####(3)'
'VA' TEXTE/

'IA' UPAWP/    ('R2) := R7          *** DUMMY!
' A'         ('R2) := R3          ***
' A'         ('R2) := R4          ***
' A'         ('R2) := R5          ***

' A'         *** G6               *** RELATIVE ADRESSE (AB 0)
' A'         *** G7               *** AKTUELLER WERT

' A'         R3 := <7+TEXTA>             ***
' A'         R4 := 'H=8000'              ***
' A'         R5 := G6                    ***
' A'         $BIHEX;                     ***                  N
' A'         'IA'                                             B
            ='H=E000'                             B
             ***                                             B

' A'         R3 := <10+TEXTA>            ***
' A'         R4 := 'H=0000'              ***
' A'         R5 := G7                    ***
' A'         $BIHEX;                     ***                  N
' A'         'IA'                                             B
            ='H=E000'                             B
             ***                                             B

' A'         R5 := (R2')          ***
' A'         R4 := (R2')          ***
' A'         R3 := (R2')          ***

' A'         :SP (R2')            ***

          'ENDE'
```

 © IPKS GmbH 2007

4.2.5.4 Promea MX & BDE Terminal (ES 1)

Promea-MX

DEVICE = *ioadr*,EAMX,"*pfadname*" [,*flags*,*verz1*,*verz2*,*verz3*]

Mit diesem Parameter wird die Emulation einer PROMEA MX über eine beliebige serielle Schnittstelle definiert. Diese serielle Schnittstelle muss als COMxx unter Windows ansprechbar sein.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

EAMX Gerätebezeichnung

pfadname Pfadname der entsprechenden seriellen Schnittstelle unter Windows. Der *pfadname* wird in der Form "COMx" bzw. "\\.\COMxx" notiert.
Für EAMX kann die serielle Schnittstelle auch auf einem entfernten Rechner sein. Der *pfadname* wird in der Form “REM:x“ angegeben. Der Parameter x steht für die Portnummer des ersten (von zwei) TCP/IP-Sockets, den der Geräteprozess auf dem M2000 Rechner einrichtet (siehe Kapitel: Schnittstellenkonzentrator).

flags pabc (Parametrierung Hexa)

a=0 Keine UNKLAR Erkennung
a=1 UNKLAR Erkennung bei TTY Schnittstelle (Brücke zwischen RxD und CTS erforderlich)
a=2 DSR Abfrage vor Ausgabe auf V.24 Schnittstelle
b=0 Schnittstelle kann überparametriert werden
b=1 Schnittstelle vor Überparametrierung schützen
b=2 Schnittstelle vor Rücksetzen schützen (octs =... und odsr =... des *mode* Kommandos wirksam)
b=3 Schnittstelle vor Rücksetzen und Überparametrieren schützen
c=0 DISIT PB ist voreingestellt
c=1 GS01 PB ist voreingestellt
c=2 Anschluss ' alte DISIT-Geräte ' (kein XON/XOFF)
p=0 Puffergröße 10 kB
p=1 Puffergröße 128 kB

verz1 Verzögerungszeit in ms vor dem Senden von Daten (falls das angeschlossene Gerät zu langsam ist).
verz2 Verzögerungszeit in ms nach dem Senden von Daten.
verz3 Verzögerungszeit in ms vor dem Empfang von Daten

Hinweis Diese Betriebsart (*flag* c=2) ist dann notwendig, wenn DISIT-Geräte, die ursprünglich an R-Systemen angeschlossen waren, jetzt an der M-Emulation betrieben werden sollen.

Wichtig ! Drucker, die im M-System als MX-Geräte generiert sind, sollten auch in der Datei MPAR.SYS als EAMX-Geräte parametriert werden; ansonsten werden Fehlermeldungen wie:
'ERROR: Anzahl Interruptvektoren'
'ERROR: DEVPRB, SWCTR unbekannt'
ausgegeben.

Beispiele

PROMEA MX Schnittstelle an einer lokalen COM Schnittstelle

```
DEVICE=8202,EAMX,"\\.\COM2",1101,000,000;
DEVICE=A302:1,EAMX,"\\.\COM128",0012,000,000;
```

PROMEA MX Schnittstelle an einem entfernten Windows System

```
DEVICE=8202,EAMX,"REM:24000",101,000,000;
```

MSHUP **DEVICE** = *ioadr*,MSHUP,"*pfadname*"[,*flags*];

Mit diesem Parameter wird die Emulation eines Spezialmoduls (MSHUP) für PROMEA MX über eine beliebige serielle Schnittstelle definiert. Diese serielle Schnittstelle muss als COMxx unter Windows ansprechbar sein.
MSHUP (*sic_mshp.exe*) emuliert das Spezialmodul für Promea MX zur Ansteuerung der Hupe in MADAM-S Systemen.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl)

MSHUP Gerätebezeichnung

pfadname Pfadname der entsprechenden seriellen Schnittstelle unter Windows. Der *pfadname* wird in der Form "COMx" bzw. "\\.\COMxx" notiert.

Hinweis Die Ausgabe erfolgt über die RTS-Leitung der seriellen Schnittstelle.

Zustand RTS – Signal:

flags (Parametrierung Hexa)

		Ruhezustand	Ausgabeimpuls
0000		aus	ein
8000	Ausgabe invertiert	ein	aus

Beispiele Spezialmodul an einer lokalen COM Schnittstelle

```
DEVICE=8040,MSHUP,"\\.\COM7";
```
keine *flags* oder 0000

```
DEVICE=8048,MSHUP,"COM2",8000;
```
mit invertierter Ausgabe

 © IPKS GmbH 2007

ES **DEVICE** = *ioadr*,EAES,"*pfadname*"[,*flags*[,*test*]]

Dieser Parameter definiert den Anschluss eines BDE-Terminals der Terminalfamilie ES 1 über serielle Schnittstelle.

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).

EAES Gerätebezeichnung

pfadname Bei Verwendung der normalen seriellen Schnittstelle ist der Pfadname "\\.\COMx"

flags Parametrierung (Hexa)
Bit 0 - 3: Parametrierung serielle Schnittstelle
=0: Die Schnittstelle ist vor dem Start der Emulation mit dem 'mode'-Kommando entsprechend zu parametrieren.
=1: Übertragungsrate: 1200 (Grundeinstellung Terminal)
Datenbits: 7
Stopbits: 2
Parität: Even
=2: Übertragungsrate: 2400
Datenbits: 7
Stopbits: 2
Parität: Even
=3: Übertragungsrate: 4800
Datenbits: 7
Stopbits: 2
Parität: Even
=4: Übertragungsrate: 9600 (Einstellung bei Hella)
Datenbits: 7
Stopbits: 2
Parität: Even

Bit 4: Eingabemodus
=0: Eingabe nicht transparent
=1: Eingabe transparent

Bit 5: Ausgabemodus
=0: Ausgabe nicht transparent
=1: Ausgabe transparent

test Nur für Testzwecke
=0: (kein Test) Normalbetrieb
=1: Ausgabe Baudrate und Ein-/Ausgabemodus nach 'Parametrieren'

Beispiel
```
DEVICE=8202,EAES,"\\.\COM2",34;
DEVICE=A302:1,EAES,"\\.\COM188",34;
```
(Eingabe transparent und Ausgabe transparent (3), 9600 Baud (4))

5 Konfigurationsdatei für SIC-R

5.1 Allgemeine Einträge

CPU-Typ

CPU = *typ*

Die Angabe des CPU-Typs *typ* ist vor allem für hardwarenahe Programme und für die Genauigkeit von Rechenergebnissen wichtig.
Folgende CPU-Typen werden unterstützt: R10, R10V, R20, R30

Beispiel
```
cpu = r10v;
```

Hauptspeicherausbau

MAXMEM = *size*

size wird als Dezimalzahl angegeben und bezeichnet die Größe des Hauptspeichers in KW im Bereich von 64 bis 1024. Dies entspricht dem Maximalausbau von 2 MB des Ursprungssystems. Der Parameter kann dem Generierprotokoll des Originalsystems entnommen werden.

Beispiel
```
maxmem = 512;
```

Hauptspeicher-RettDatei

IMAGE = "*pfadname"[,mode]*

Mit diesem Parameter wird eine Windows-Datei definiert, in die bei Programmende der Hauptspeicherinhalt des emulierten SICOMP R Systems geschrieben wird (es wird ein batteriegepufferter Hauptspeicher emuliert). Programmende heißt, dass die Emulation gezielt beendet wird.
(M2000Register `Monitor` Button `Stoppen`)

pfadname Vollständiger Pfadname einer Windows-Datei. Sie wird implizit eingerichtet.

mode Bearbeitungsmodus für die ImageDatei.

- NEW (auch Default). Die *image* Datei wird beim Programmende immer neu unter dem gleichen Namen angelegt. Eine bereits bestehende *image* Datei wird zuvor gelöscht.
- ALTx Mit der Angabe ALTx (x=1 bis 9) werden x *image* Dateien angelegt, die alternierend überschrieben werden. Die *image* Dateien haben die Namen *dateiname.im1* ..... *dateiname.imx*. Im Kopf jeder *image* Datei werden Datum/Uhrzeit und eine Konsistenzkennung hinterlegt. Beim Beginn des Schreibens wird eine Inkonsistenzkennung eingetragen, wenn der HSP Inhalt vollständig abgespeichert ist, wird eine Konsistenzkennung mit Datum/Uhrzeit gesetzt. Überschrieben wird immer die älteste oder eine inkonsistente Datei.

Wird die Emulation wieder gestartet und die Hauptspeicher RettDatei ist vorhanden, so wird der Hauptspeicher des emulierten SICOMP R Systems aus der Hauptspeicher RettDatei vorbelegt. Dem ORG werden die entsprechenden Kennungen für einen batteriegepufferten Wiederanlauf übergeben. Damit kann eine Rekonstruktion von DVS-Dateien mittels DFCONS erfolgen.

Beispiel
```
IMAGE = "\user\pcsic\sysrett.v1";
```


© IPKS GmbH 2007

Urladegerät

BOOT = *hexboot*

hexboot ist eine Hexadezimalzahl (4stellig), die das Urladegerät angibt. Sie entspricht der Einstellung der Schalter am Wartungsfeld.

Beispiel

```
BOOT = 0025;
```

Gleitpunktprozessor deaktivieren

NOFPP

Der Gleitpunktprozessor führt die arithmetischen Verknüpfungen für 32-Bit-Festpunkt- sowie 32- und 64-Bit-Gleitpunktoperationen aus und ist standardmäßig in der Emulation aktiviert. Für spezielle Anforderungen ist es möglich, die Funktion des Gleitpunktprozessors mit dieser Anweisung zu deaktivieren.
(NOFPP = No Floating Point Processor)

Beispiel

```
NOFPP
```

MEC-28R

MECR = *anr*

Beim Betrieb einer MEC-28R Anschaltung ist dieser Parameter erforderlich. Für jede Anschaltung ist eine eigene Angabe erforderlich.

anr Anschlussstellennummer der Anschaltung (Dezimalzahl)

Beispiel

```
MECR = 7;
```

5.1.1 Sichtgeräte

Sichtgeräte seriell angeschlossen

DEVICE = *anr,gnr,keyword,*"*pfadname*"[,*upar*[,*kennung*[,*verz*]]]

Mit diesem Parameter wird der Anschluss eines Sichtgerätes über eine serielle Schnittstelle (COMxx) definiert.

anr,gnr Anschlussstellen-/Gerätenummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

keyword Folgende Sichtgeräte können hier angegeben werden:
3974 Anschluss an Anschaltung C71458-A6404-A..
3974R Anschluss an PROMEA-Baugruppe
3974M Anschluss an PROMEA-Baugruppe

Für die Terminal-Emulationen muss *keyword* wie folgt verwendet werden:
3974R für TE2000 AX oder TE-ALPHA
3974M für TE2000 DX oder TE-DISIT

pfadname Bezeichnung der unter Windows definierten seriellen Schnittstelle. Der *pfadname* wird in der Form "COMx" bzw. "\\.\COMxx" notiert.

upar Modus 1: Überparametrierung (Hexa)
= 0: Überparametrierung möglich
= 1: Schnittstelle vor Überparametrierung schützen (default)
Modus 2: Unklar-Erkennung (Hexa)
= 100: Unklar-Erkennung bei TTY Schnittstelle
(Brücke zwischen RxD und CTS erforderlich)
= 200: DSR Abfrage vor Ausgabe auf V.24 Schnittstelle
Die Modi 1 und 2 können mit ODER verknüpft werden

kennung Freigaben im externen/internen Dongle
Kennung **IPKS**: Funktionsfreigabe der (alten) TE-ALPHA/TE-DISIT Terminal-Emulation im zentralen M2000 Dongle.
Kennung **TE2000**: Funktionsfreigabe der TE2000AX/DX Terminal-Emulation im zentralen M2000 Dongle.
Kennung **NONE**: Funktionsfreigabe im externen Dongle erforderlich.

verz Verzögerungszeit (Millisekunden) in 32-Bit-Darstellung.
Die höherwertigen 16 Bit werden als Eingabeverzögerung, die niederwertigen als Ausgabeverzögerung interpretiert.

Hinweise

- Die Schnittstelle ist vorher mit dem Windows *mode*-Kommando entsprechend zu parametrieren.
- Die Flusskontrolle wird von M2000 auf XON/XOFF=93/94h eingestellt.
- Siehe Tabelle 1 in diesem Kapitel (SichtgeräteTypen)
- Bei fehlenden Optionen in der Parameterzeile müssen Kommas trotzdem eingetragen werden.

Beispiel

```
DEVICE=1,0,3974R,"\\.\COM3",100,IPKS,0x00640032;
```

Terminalanschluss (TTY) mit unklar Erkennung, zentrale Funktionsfreigabe, EingabeVerz.=100ms, AusgabeVerz.=50ms.

```
DEVICE=1,1,3974M,"\\.\COM4",1,NONE,0x00640032;
```

DISIT-Version, Überparametrieren nicht möglich, externer Dongle, EingabeVerz.=100ms, AusgabeVerz.=50ms.

 © IPKS GmbH 2007

Terminal-Emulation lokal

DEVICE = *anr,gnr,keyword,"pipename"*

Mit diesem Parameter wird eine Terminal-Emulation als lokale Windows Applikation aktiviert.

anr,gnr Anschlussstellen-/Gerätenummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

keyword Folgende Typangaben sind zulässig:
3974RT für eine lokale Konsole
3974MT für eine lokale DISIT-Emulation

pipename Eindeutiger Name für die Konsolen-Applikation. Unter diesem Namen wird eine *named pipe* angelegt, die damit systemweit eindeutig sein muss.
Zusätzlich erscheint dieser Name als Fensterüberschrift.

Hinweise

- Bei Verwendung im Fenster sollte für den Monitor mindestens eine Auflösung von 800x600 (16/256 Farben) eingestellt sein.
- Dieser Parameter kann mehrfach definiert werden, um mehrere Sichtgeräte zu emulieren. Für jedes Gerät wird dann ein eigenes Fenster angelegt.
- Zur besseren Performance sollte eine leistungsfähige Grafikkarte (PCI) eingesetzt werden.
- Der Leistungsumfang dieser Geräteemulation ist in Kapitel 6.1 kurz beschrieben. Eine ausführliche Beschreibung enthält das Benutzerhandbuch: **TE2000 Terminal-Emulation**.
- Verwenden Sie in dem Pipenamen bitte keine Sonderzeichen (wie - \ , / . ? usw.). Es kann sonst vorkommen, daß mehrere Terminalemulationen mit der gleichen Pipe arbeiten.
  Beispiel: statt **A-DSSK002** schreiben Sie bitte **A_DSSK002**

Beispiel

```
DEVICE=1,1,3974RT,"DSSE1";
```

DISIT-Emulation TE2000 DX **über Netzwerk**

DEVICE = *anr,gnr*,3974MNET,"*socketnr[/t]"[*,TE2000*[,timer]]*

Mit diesem Parameter wird der Betrieb einer DISIT Terminal-Emulation TE2000 DX über Netzwerk definiert.

DISIT-Emulation TE-DISIT **über Netzwerk**

DEVICE = *anr,gnr*,3974MNET, "*socketnr[/t]"[*,IPKS*[,timer]]*

Mit diesem Parameter wird der Betrieb einer (alten) Terminal-Emulation TE-DISIT über Netzwerk definiert.

anr,gnr Anschlussstellen-/Gerätenummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

socketnr Socket-Nummer, über die M2000 mit der Terminalemulation kommunizieren soll. Die Socket Nummer muss beim Start der entfernten Terminalemulation als Startparameter mit übergeben werden. Socket-Portnummer, gemäß Administration.
Empfohlener Bereich: >10000
Parameter **/ t** siehe nächste Seite.

TE2000 Die Kennung TE2000 legt fest, dass diese Terminal-Emulation im zentralen M2000 Dongle freigegeben ist.

IPKS Die Kennung IPKS legt fest, dass diese Terminal-Emulation im zentralen M2000 Dongle freigegeben ist.

timer Zeit in ms um Ein-/Ausgaben bei langsamen Sichtgeräten (z.B. ES05) oder Anwenderprogrammen zu verzögern.

Beispiel

```
DEVICE=1,2,3974MNET,"20001";
DEVICE=2,1,3974MNET,"20011/30",TE2000,1500;
```

© IPKS GmbH 2007

Terminal-Emulation **DEVICE** = *anr,gnr*,3974RNET,"*socketnr[/t]*"*[*,TE2000*[*,*timer]]*
TE2000 AX **DEVICE** = *anr,gnr*,3974RNETS,"*socketnr[/t]*"*[*,TE2000*[*,*timer]]*
über Netzwerk

Mit diesem Parameter wird der Betrieb einer Terminal-Emulation TE2000 AX über Netzwerk definiert.

Terminal-Emulation **DEVICE** = *anr,gnr*,3974RNET,"*socketnr[/t]*"*[*,IPKS*[*,*timer]]*
TE-ALPHA
über Netzwerk

Mit diesem Parameter wird der Betrieb einer (alten) Terminal-Emulation TE-ALPHA über Netzwerk definiert.

anr,gnr Anschlussstellen-/Gerätenummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

socketnr Socket-Nummer, über die M2000 mit der Terminalemulation kommunizieren soll. Die Socket Nummer muss beim Start der entfernten Terminalemulation als Startparameter mit übergeben werden. Socket-Portnummer, gemäß Administration.
Empfohlener Bereich: >10000
Eine aufgebaute Verbindung zwischen M2000 und der Terminalemulation wird zeitüberwacht. Wenn innerhalb von t (default 180) Sekunden kein Datenverkehr stattfindet, baut M2000 die Verbindung wieder ab. Die Überwachungszeit *t* kann mit der Socketnummer parametriert werden (siehe Beispiele).

TE2000 Die Kennung TE2000 legt fest, dass diese Terminal-Emulation im zentralen M2000 Dongle freigegeben ist. (TE2000AX/DX)

IPKS Die Kennung IPKS legt fest, dass diese Terminal-Emulation im zentralen M2000 Dongle freigegeben ist. (TE-ALPHA/TE-DISIT)

timer Zeit in ms um Ein-/Ausgaben bei langsamen Sichtgeräten (z.B. ES05) oder Anwenderprogrammen zu verzögern.

Anmerkung Wird 3974RNETS parametriert, dann führt ein **BFA Rücksetzen** nicht zu einem Abbau der Socketverbindung.

Beispiel

```
DEVICE=1,3,3974RNETS,"20011";
DEVICE=2,1,3974RNET,"20011/30",TE2000,1500;
```

Tabelle 2 Sichtgerätetyp - keyword für die Parametrierung in *rpar.sys*

Sichtgeräte Typ	**keyword** serieller Anschluss	lokales Terminal	Netzwerk-Terminal
3974	3974	—	—
3974R	3974R	—	—
3974M	3974M	—	—
DS074	3974R	—	—
DS075	3974R	—	—
DS075 DISIT	3974M	—	—
TE2000 AX	3974R	3974RT	3974RNET
TE2000 DX	3974M	3974MT	3974MNET
TE-KONS	—	3974RT	—

© IPKS GmbH 2007

Anmerkung Sind im Rechner zwei (oder mehr) Netzwerkkarten installiert, kann einem Sichtgerät auch gezielt die IP-Adresse (oder der Alias-Name) der jeweiligen Karte zugeordnet werden. Der Alias-Name muss in der Datei *hosts* in **systemdir***system32**drivers**etc*\... eingetragen sein. Die zugehörige Parameterzeile ist folgendermaßen aufgebaut:

DEVICE = *anr,gnr,*3974RNET*, ipadr*:*port,kennung*
DEVICE = *anr,gnr,*3974MNET*,aliasname:port,kennung*

Beispiel

```
DEVICE=2,0,3974RNET,192.168.2.10:20010,IPKS;
```

```
DEVICE=2,0,3974RNET,NetCard-LAN1:20010,IPKS;
```
Verbindung wird über Netzwerkkarte 1 aufgebaut.

```
DEVICE=2,1,3974RNET,192.168.2.13:20013,TE2000;
```

```
DEVICE=2,1,3974MNET,192.168.2.13:20013,IPKS;
```

```
DEVICE=2,1,3974MNET,NetCard-LAN2,20013,TE2000;
```
Verbindung wird über Netzwerkkarte 2 aufgebaut.

Hinweis Eine ausführliche Beschreibung enthält das Benutzerhandbuch:
TE2000 Terminal-Emulation

5.1.2 Speichermedien

Belegmodus der Plattenlaufwerke festlegen

mode = share

Mit dem Parameter MODE wird der Belegungsmodus der Platten gewählt. (exclusiv/shared)
Fehlt der Parameter, ist der Belegmodus '*exclusiv*' eingestellt (default).
Die Verwendung des Parameters ermöglicht den Zugriff auf die Platten aus beiden Systemumgebungen.

Plattenlaufwerk **3941,3942,3945,3946, 3947,3948,3949**

DEVICE = *anr,gnr,keyword,*"*pfadname*"[*,flags*]

Mit diesem Parameter wird ein Gerät eines Plattenlaufwerks zur Emulation durch eine Windows Datei definiert.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

keyword

3941	bezeichnet ein Gerät mit 9744 Sektoren
3942	bezeichnet ein Gerät mit 48720 Sektoren
3945A	bezeichnet ein Gerät mit 2048 Sektoren
3945B	bezeichnet ein Gerät mit 4096 Sektoren
3945C	bezeichnet ein Gerät mit 8192 Sektoren
3946	bezeichnet ein Gerät mit 8192 Sektoren
3947	bezeichnet ein Gerät mit 15616 Sektoren
3947A	bezeichnet ein Gerät mit 15616 Sektoren
3947H	bezeichnet ein Gerät mit 1034240 Sektoren
3948	bezeichnet ein Gerät mit 64640 Sektoren
3948A	bezeichnet ein Gerät mit 64640 Sektoren
3948B	bezeichnet ein Gerät mit 61408 Sektoren
3949	bezeichnet ein Gerät mit 25856 Sektoren
3949A	bezeichnet ein Gerät mit 25856 Sektoren
3949B	bezeichnet ein Gerät mit 25856 Sektoren
3949C	bezeichnet ein Gerät mit 25856 Sektoren

pfadname Pfadname der entsprechenden Windows-Datei. Er muss immer vollständig angegeben werden. Eine nicht vorhandene Datei richtet der Emulator ein (Länge entsprechend dem Gerätetyp). Sie kann dann mit INIT formatiert und eingerichtet werden.

flags Parametrierung (Hexa)
= 1 WRITE THROUGH
Der Parameter gewährleistet, dass veränderte Daten auf die Platte geschrieben werden und nicht im CACHE verbleiben;
Option sinnvoll bei DVS Datenträgern.

Beispiel

```
DEVICE=5,2,3948, "c:\sicomp\platten\SYSR_PS2.sic",1;
```


© IPKS GmbH 2007

Diskettenlaufwerk **DEVICE** = *anr,gnr,keyword,"pfadname"*

Dieser Parameter definiert ein Diskettenlaufwerk.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

keyword
- 3943 bezeichnet ein Gerät mit 256256 Bytes
- 3944 bezeichnet ein Gerät mit 1021696 Bytes
- FLOP bezeichnet ein Gerät mit 1228800 Bytes

Der Betrieb eines PC-Diskettenlaufwerks (3,5", 1.44MB oder 5.25", 1.2MB) erfolgt ohne Kompatibilität zu SICOMP. Dieses Diskettenlaufwerk kann nur zur Datensicherung und zum Datenaustausch zwischen mehreren Emulationen genutzt werden. Im emulierten System wird das Diskettenlaufwerk wie ein 3943/3944 Laufwerk genutzt (auch mit *INIT*). Eine solche Diskette ist aber auf einem SICOMP Rechner nicht verwendbar.

pfadname Der Pfadname heißt: \\.\A:
\\.\B:

Beispiel
```
DEVICE=3,0,3944,\\.\A:;
```

Erweiterter **Floppy Zugriff**

Mit M2000 können (unter DOS formatierte) Disketten-Dateien bearbeitet werden, die mit dem folgenden SICOMP R Datenträger kompatibel sind:

- 8-Zoll Floppy Ersatzsystem der EDS Systemtechnik mit 908-PC Adapterkarte und dem Programm BDK.

In der Parameterdatei *rpar.sys* ist dieser Floppy Zugriff folgendermaßen zu parametrieren:

Als Pfadname wird statt "\\.\A:" nunmehr angegeben z.B. "A:\EKTEST".
M2000 sucht beim Zugriff auf die Diskette die angegebene Datei (hier EKTEST). Wenn die Datei auf der Floppy vorhanden ist, wird sie verwendet. Wird die angegebene Datei (hier EKTEST) nicht gefunden, so sucht M2000 eine Datei auf der Diskette mit der passenden Länge. Wird eine entsprechend lange Datei gefunden, so wird diese verwendet. Wird keine entsprechend lange Datei gefunden, so wird eine Datei mit dem parametrierten Namen (hier EKTEST) auf der Diskette eingerichtet. Der in der Parametrierdatei angegebene Name (hier "EKTEST") ist also nur dann relevant, wenn M2000 eine Datei neu einrichten soll. Vorhandene Dateien können beliebige Namen haben.

© IPKS GmbH 2007

Magnetband-Kassetten-Laufwerk **MK82**

DEVICE = *anr,gnr,*MK82*,"pfadname"*

Dieser Parameter ermöglicht den Betrieb eines MK82 kompatiblen Magnetbandkassettenlaufwerks Tandberg Data Panther TDC 3660 bzw. 3820 zum Lesen und Schreiben von 150 MB-Kassetten.
Magnetbandkassetten eines MK80/81 Gerätes mit 60MB können nur gelesen werden. Zusätzlich ist noch eine MECR-Anweisung erforderlich.

anr Anschlussstellennummer des Gerätes (Dezimalzahl)

gnr irrelevant

pfadname Pfadname des entsprechenden Windows-Gerätes

Beispiel

```
DEVICE=7,0,MK82,\\.\Tape0;
MECR=7;
```

Optische Platte OP11

DEVICE = *anr,gnr,*OP11*,"pfadname",scsi-id*

Dieser Parameter definiert die Emulation einer wiederbeschreibbaren optischen Platte OP11 über das Laufwerk SONY RMO-S580.
Zusätzlich ist noch eine MECR-Anweisung erforderlich.

anr Anschlussstellennummer des Gerätes (Dezimalzahl)

gnr Gerätenummer (Dezimalzahl)

pfadname Der Pfadname heißt "\\.\PhysicalDrive%"
% ist die Disknummer innerhalb Windows, beginnend mit 0 (siehe Windows Festplatten Manager).

scsi-id SCSI-Laufwerksnummer (Dezimalzahl, 0..7) des emulierten Laufwerks im Ursprungssystem, nicht im PC.

Beispiel

```
DEVICE=7,0,OP11,"\\.\PhysicalDrive1",3;
MECR=7;
```

Magnetwechselplatte **WP256**

DEVICE = *anr,gnr,*WP256*,"pfadname",scsi-id*

Dieser Parameter definiert die Emulation eines Laufwerks für Magnetwechselplatten (z. B. SYQUEST). Zusätzlich ist noch eine MECR-Anweisung erforderlich.

anr Anschlussstellennummer des Gerätes (Dezimalzahl)

gnr Gerätenummer (Dezimalzahl)

pfadname Der Pfadname heißt "\\.\PhysicalDrive%"
% ist die Disknummer innerhalb Windows, beginnend mit 0

scsi-id SCSI-Laufwerksnummer (Dezimalzahl, 0..7) des emulierten Laufwerks im Ursprungssystem, nicht im PC.

Beispiel

```
DEVICE=7,0,WP256,"\\.\PhysicalDrive1",1;
MECR=7;
```

 © IPKS GmbH 2007

5.1.3 Drucker

Druckerbetrieb an der **Parallel-Schnittstelle**

DEVICE = *anr,gnr,*DRUA*,"pfadname"[,flags[,timer]]*

Mit diesem Parameter wird das Gerät *DRUA* auf einen an der parallelen Schnittstelle installierten Drucker umgeleitet.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pfadname Pfadname des entsprechenden Windows-Gerätes, in der Regel "LPT1"

flags Kennung:

3915	Drucker 3915 (ES902 oder PROMEA)
3915B	Drucker 3915 (SIVAREP-B)
3916	Drucker 3916 (PROMEA)
3916B	Drucker 3916 (SIVAREP-B)
3916E	Drucker 3916 (ES902)
3918	Drucker 3918
3919	Drucker 3919
CRLF	Ersetzen von LF am Transferende durch CR/LF

timer Wert in Sek. für WRITE Überwachung (Default: 20 Sec)

Beispiel

```
DEVICE=1,1,DRUA,"LPT1";
```

Wichtig! Für die parallele Schnittstelle darf in Windows kein Druckertreiber installiert bzw. aktiviert sein.

Druckausgabe über **Windows Druckertreiber**

DEVICE = *anr,gnr,*DRSPOOL*,"pfadname"*[*,flags*[*,timer*]]
DEVICE = *anr,gnr,*DRSPOOLF,"*pfadname*"[*,flags*[*,timer*]]

Mit diesem Parameter wird das Gerät DRUA an einen Windows Druckertreiber weitergeleitet. Dabei wird mit der ersten Ausgabe ein Dokument geöffnet und dieses wieder geschlossen, wenn 5 Sek. keine weitere Ausgabe erfolgt. Eine erneute Ausgabe nach mehr als 5 Sek. führt zu einer weiteren Dokument-Eröffnung.
DRSPOOLF schließt Ausgaben mit einem zusätzlichen Steuerzeichen FF (FormFeed) ab. (bei einigen LaserDruckern erforderlich)

Die Parameter entsprechen denen der parallelen Ausgabe, s.o. (DRUA parallel).

Beispiel

```
DEVICE=1,1,DRSPOOL,"LPT1";
DEVICE=2,1,DRSPOOL,"\\DRSERVER\DRUA00",10;
DEVICE=2,1,DRSPOOLF,\\192.168.1.7\DRUA10;
```

Anmerkung Bei Ausgabe auf einen "NetzDrucker" bedeutet:

DRSERVER	=	Name des Druck(er)Servers
192.168.1.7	=	IP Adresse des Druck(er)Servers
DRUA00	=	lokaler Freigabename des "NetzDruckers"
DRUA10	=	lokaler Freigabename des "NetzDruckers"

Druckausgabe in **Windows Datei**

DEVICE = *anr,gnr,*DRUA*,"pfadname"[,flags]*

Mit diesem Parameter wird die Druckausgabe in eine Windows Datei umgeleitet. Es wird ab Dateianfang geschrieben, eine evtl. existierende Datei wird zuvor gelöscht.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pfadname Vollständiger Pfadname einer Windows Datei in die umgeleitet werden soll. Die Datei wird implizit eingerichtet.

flags Kennung (siehe DRUA)

Beispiel

```
DEVICE=1,1,DRUA,"drua0";
```

Druckerbetrieb (seriell) mit **DR202**

DEVICE = *anr,gnr,*DR202*,"pfadname"[,flags[,timer]]*

Dieser Parameter ermöglicht den Betrieb eines an der seriellen Schnittstelle angeschlossenen Druckers DR202.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pfadname Pfadname der entsprechenden seriellen Schnittstelle unter Windows

flags Parametrierung (Hexa)
= 1 BREAK Erkennung

timer Wert in Sek. für WRITE Überwachung (Default = 20Sec)

Beispiel

```
DEVICE=1,1,DR202,"COM7";
```


© IPKS GmbH 2007

Druckerbetrieb (seriell)
alle Drucker

DEVICE = *anr,gnr,*DRCOM*,"pfadname"[,typflags[,time1][,time2]]*

Dieser Parameter ermöglicht den Betrieb aller an der seriellen Schnittstelle angeschlossenen Drucker.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pfadname Pfadname der entsprechenden seriellen Schnittstelle unter Windows

typflags Parametrierung (Hexa)
Hier kann entweder:
der Druckertyp in der Form ‘’typ’’ ,
der String ‘’CRLF’’, oder
die Flags:
= 1 Break-Erkennung bei TTY
= 2 Break-Erkennung bei V.24
= 4 ready/busy
= 8 seriellen Port nicht parametrieren, das MODE-Kommando ist gültig
= 10 wie Parameter(string) CRLF

time1 Wert in Sek. für WRITE Überwachung (Default = 20Sec)

time2 Verzögerungszeit in ms vor dem Senden von Daten (falls das angeschlossene Gerät zu langsam ist)

Beispiel

```
DEVICE=1,1,DRCOM,"COM7",3915,,100;
```

© IPKS GmbH 2007

Druckausgabe in **Windows Datei** (parametergesteuert)

DEVICE = *anr*,*gnr*,DRSPOTX,"*file*"

Mit DRSPOTX wird die Druckausgabe parametergesteuert in eine Windows Datei umgeleitet.

anr	Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)
gnr	Gerätenummer gemäß ORG-Generierung (Dezimalzahl)
file	(Pfad)Name der Parameterdatei.

Anmerkung

Beim Erstellen der Parameterdatei müssen folgende Regeln beachtet werden:
Kommentare beginnen mit einem Semikolon.
Die Parameternamen SUPPRESS, CHANGE, START, STOP können jeweils bis zu 255 mal verwendet werden, ein String kann bis zu 64 Zeichen lang sein, alle Zeichen sind erlaubt. Die übrigen Parameternamen dürfen nur einmal verwendet werden, bei Mehrfachangaben gilt jeweils die letzte Definition.
In den Zeichenketten "string" können Sonderzeichen, wie unten beschrieben, als Hexamuster dargestellt werden:
Die beiden einem Backslash(****) nachfolgenden Zeichen werden als Hexamuster interpretiert. Das Zeichen **** selbst kann nur als \5d dargestellt werden.
Das Zeichen **"** kann nur als \22 dargestellt werden.
In einem Ausgabesatz wird nur ein START- oder STOPstring interpretiert.
Eine Änderung der Parameterdatei im laufenden Betrieb wird erst nach einem Neustart des Emulators wirksam oder durch das Auslösen der Funktion "Gerät rücksetzen" mit Hilfe des Programms *TWR.EXE*. Vor der Funktion wird ggfs. eine gerade eröffnete Datei geschlossen.
Zeitgesteuerte (einmalig oder wiederkehrend) Aktivitäten müssen mit dem Windows Kommando **at** *zeit befehl* realisiert werden. Dazu muss der Zeitplandienst (SCHEDULE+) von Windows gestartet sein. (**net start** schedule)

Beispiel

at 23:00 netcopy.bat

Die dynamischen Funktionen werden in der Reihenfolge:

1. SUPPRESS:	alle zu löschenden Zeichen werden entfernt
2. CHANGE:	alle nach dem Löschen noch vorzunehmenden Ersetzungen werden ausgeführt
3. START:	es wird geprüft, ob der Datensatz nach den Löschvorgängen und nach den Ersetzungen eine "Startzeichenfolge" enthält
4. STOP:	es wird geprüft, ob der Datensatz nach den Löschvorgängen und nach den Ersetzungen eine "Stopzeichenfolge" enthält

bearbeitet.

Parameternamen:

PATH [*drive*:[*path*]]	; Pfadname unter dem die zu erzeugenden Dateien abgelegt werden.
SPOTTIME *sec*	; Einstellen der Überwachungszeit (default: 0, keine Überwachungszeit).
DOSNAMES	; Verwenden von DOS-kompatiblen Dateinamen,
NTNAMES	; ...nicht DOS-kompatiblen Dateinamen (default).
STARTNAMES *n*	; *n* Zeichen, die START "*string*" folgen, bilden den Anfang des Dateinamen. (z...z_JJJJMMTThhmmss.anrgnr)
LFDNAMES	; Bilden der Dateinamen als ' laufende Nummer '
EXTENSION x....x	; Angabe einer frei wählbare (Datei)Extension
SHARE	; ExklusivBelegStatus der AusgabeDateien wird aufgehoben.
SUPPRESS "*string*"	; Entfernen von Zeichen(folgen) vor der Ausgabe.
CHANGE "*string1*" "*string2*"	; Ersetzen von Zeichen(folgen) vor der Ausgabe; *string1*/*string2* können verschieden lang sein.
START "*string*"	; Definition einer "Startzeichenfolge"
STOP "*string*"	; Definition einer "Stopzeichenfolge"
WINDOW "*name*"	; Ausgaben zusätzlich in Fenster *name*
WINDOWT "*name*"	; `dito`, zusätzlich mit vorangestelltem Zeitstempel

© IPKS GmbH 2007

COMMAND "[*drive*:[*path*]] *file*" ; Definition eines Kommandos;

es wird nach dem Schließen einer Zieldatei abgesetzt. Dem Kommando können 5 Parameter mitgegeben werden:

der 1. Parameter bezeichnet die gerade geschlossene Datei.
der 2. Parameter enthält Kennungen wie,
Datei geschlossen nach Ereignis:
"to" *Zeitspanne*
"aa" *Startzeichenfolge*
"zz" *Stopzeichenfolge*
"bo" *ORG-Boot* oder *Drucker rücksetzen*
"nt" nach Beenden der Emulation

der 3. Parameter gibt nur das Verzeichnis an, in dem die Datei abgelegt ist.
der 4. Parameter gibt nur den Namen der gerade geschlossenen Datei an.
der 5.Parameter gibt nur die Erweiterung der gerade geschlossenen Datei an.

Beispiel

```
DEVICE=2,2,DRSPOTX,"\sicomp\spotx.par";
```

Die Parameter-Datei *spotx.par* könnte z.B. so aussehen:

```
PATH e:\temp                                                    ; ( 1 )
DOSNAMES                                                        ; ( 2 )
EXTENSION DRUA1                                                 ; ( 3 )
SPOTTIME 20                                                     ; ( 4 )
SUPPRESS "\0D\0A\03"                                            ; ( 5 )
CHANGE "\0D\0A" "\0A"                                           ; ( 6 )
START "//JOB:"                                                  ; ( 7 )
STOP   "//JEND"                                                 ; ( 8 )
STARTNAMES 7                                                    ; ( 9 )
SHARE                                                           ; ( 10 )
COMMAND "e:\temp\drspotx.bat" "drfile1" "to" "\temp" "drfile1" "txt"   ; ( 11 )
```

(1) Die Druckdateien sollen im Verzeichnis *e:\temp* abgelegt werden.
(2) Es sollen DOS-kompatible Dateinamen verwendet werden.
(3) Die frei gewählte (Datei)Erweiterung ist .DRUA1 statt *.ioadr .*
(4) Die Druckdatei wird geschlossen, wenn 20 sec keine Ausgabe erfolgt.
(5) Die Zeichenfolge CR-LF-ETX soll unterdrückt werden.
(6) Die Zeichenfolge CR-LF soll durch LF ersetzt werden.
(7) Die aktuelle Druckdatei wird geschlossen, wenn die Zeichenfolge "//JOB:" erkannt wird; eine neue Datei wird angelegt und "//JOB:" wird in die Neue übernommen.
(8) Die aktuelle Druckdatei wird geschlossen, wenn die Zeichenfolge "//JEND" erkannt wird und der String "//JEND" wird an das Ende der zu schließenden Datei geschrieben. Anschließend wird eine neue Druckdatei eröffnet, in die weitere Daten bis zum (jeweils gewünschten) Ausgabeende geschrieben werden.
(9) Die ersten 7 Zeichen, die "//JOB:" folgen (z.B. JOBCOPY), bilden den Anfang des neuen Dateinamens. z.B. *JOBCOPY_20020812135801.0202*
(10) Der (Datei)Status *geteilte Belegung* wird aktiviert, die Datei kann während der Ausgabe betrachtet werden.
(11) Die Druckdatei *drfile1.txt* wird wegen Kennung *to* nach Ablauf von Zeitspanne (SPOTTIME) geschlossen. Nach dem Schließen der Datei soll der Batch *drspotx.bat* aktiviert werden.

Anmerkung Bei SPOTTIME 0 (default) bleibt die zuletzt angelegte Datei solange offen und 'aufnahmefähig', bis andere Ereignisse (START ..., STOP ..., etc.) ein Schließen der Datei veranlassen.

Druckausgabe in ein **Windows Fenster**

DEVICE = *anr,gnr,*DRWIN*,"fenster","datei1","datei2","size"*
Mit diesem Parameter wird die Druckausgabe in ein Windows Fenster umgeleitet und zusätzlich in Dateien geschrieben.

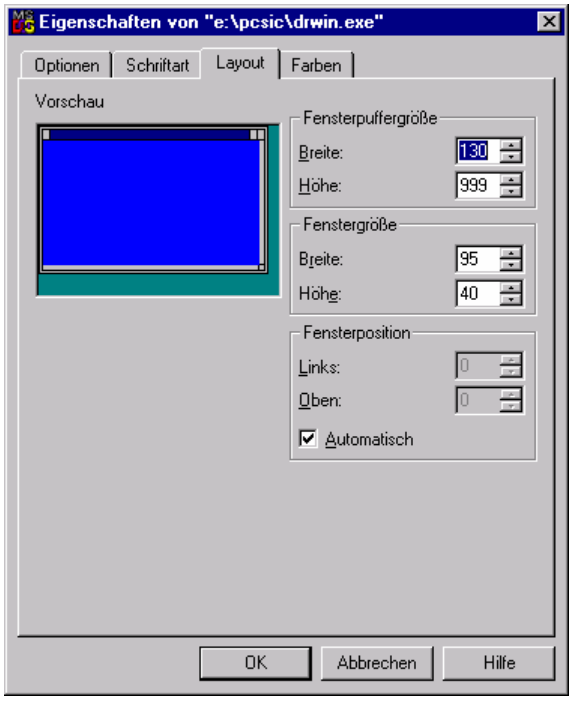
anr	Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)
gnr	Gerätenummer gemäß ORG-Generierung (Dezimalzahl)
fenster	Fenstername
datei1	erste Protokolldatei
datei2	zweite Protokolldatei
size	max. Dateigröße in Kilobyte

Beispiel

```
DEVICE=1,1,DRWIN,"DRUA0";
DEVICE=2,2,DRWIN,"DRUA","outfile1","outfile2","10";
```

Der Parameter DRWIN besagt, dass die Druckausgaben in ein Windows Fenster umgeleitet werden. Nachdem Sie die Emulation gestartet haben, erscheint jetzt auf Ihrem Bildschirm ein weiteres Fenster mit dem Namen DRUA0. Dieses Fenster können Sie konfigurieren. Klicken Sie mit der rechten Maustaste die Kopfleiste des Fensters DRUA0 an. Wählen Sie die Funktion *Eigenschaften*. Über *Layout* ⇒ *Fensterpuffergröße* - *Fenstergröße* legen Sie fest, wieviel Druckerzeilen das Fenster puffern kann (beispielsweise 999). Das Windows Fenster arbeitet jetzt wie ein Umlaufpuffer mit einer Größe von 999 Zeilen.

Wenn Sie jetzt Teile der Druckerausgaben (beispielsweise Fehlermeldungen) auf einem am PC angeschlossenen Drucker ausgeben wollen, gehen Sie wie folgt vor:
Klicken Sie mit der rechten Maustaste die Kopfleiste an und wählen Sie die Funktionen *Bearbeiten* ⇒ *Markieren*. Markieren Sie im Fenster die Zeilen, die Sie drucken wollen. Übernehmen Sie den markierten Bereich in die Zwischenablage (CTRL und C). Rufen Sie ein Textprogramm auf wie beispielsweise WRITE. Übernehmen Sie dort den markierten Bereich in ein Dokument (CTRL und V) und drucken Sie das Dokument aus.



© IPKS GmbH 2007

(Zusätzliche) Ausgabe in Protokolldateien:
Ist eine zusätzliche Ausgabe in Protokolldateien gewünscht, dann müssen die Parameter *datei1*, *datei2* und *size* angegeben werden.
(siehe Beispiel 2)
Zuerst erfolgt die Ausgabe in *datei1*. Hat diese die maximale Größe (*size**1024) überschritten, dann wird mit *datei2* weitergearbeitet. Ist diese auch voll, wird *datei1* gelöscht und neu beschrieben.
Es wird jeweils eine komplette Druckerzeile in die Datei geschrieben und erst danach auf die Größe abgefragt.
Beim Beenden der Emulation wird die letzte Einstellung in der Datei *fenster.par* (z.B. *DRUA0.par*) abgelegt.
Hat sich nach einem Neustart nichts an den Parametern geändert, dann werden die Ausgaben an die zuletzt geschriebene Protokolldatei angehängt und wie oben beschrieben fortgefahren.
Wurden Parameter geändert oder treten Probleme beim Öffnen der zuletzt geschriebenen Protokolldatei auf, dann wird mit der Ausgabe in die Datei *datei1* neu begonnen.
Beim Beenden wird eine Meldungszeile (--Ende DRWIN--) an das Ende der aktuellen Protokolldatei geschrieben.

Druckausgabe mit Datum/Uhrzeit in ein **Windows Fenster**

DEVICE = *anr*,*gnr*,DRWINT,"*fenster*"

DRWINT entspricht dem Parameter DRWIN; zusätzlich wird in jede Zeile der Druckausgabe am Zeilenanfang Datum und Uhrzeit im Format: `JJJJ-MM-TT hh:mm:ss.ms` geschrieben. Ideal für Meldedrucker.

anr	Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)
gnr	Gerätenummer gemäß ORG-Generierung (Dezimalzahl)
fenster	Fenstername

Beispiele

```
DEVICE=2,2,DRWINT,"DRUA";
```

Beispiel DRWINT

```
2002-08-1311:58:02.701 CDLI TO DRUA
2002-08-13 11:58:02.701 TESTCD 28 CA CBUET 65271 42
```

Hinweis

Werden statt `DRWIN/DRWINT` die Parameter `DRWINMIN/DRWINTMIN` eingetragen, dann werden die DruckerFenster – minimiert – aktiviert.

5.1.4 Kopplungen

Rechnerkopplung **DUST3964** über eine beliebige serielle Schnittstelle

DEVICE=*anr*,gnr,3964,"*pfadname*",*p1[,p2[,p3[,p4]]]*

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DUST3964 über eine beliebige serielle Schnittstelle definiert. Diese serielle Schnittstelle muss als COMxx unter Windows ansprechbar sein.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pfadname Pfadname der entsprechenden Windows-Geräteschnittstelle, d.h. "COMx" bzw. "\\.\COMxx". Die Schnittstelle muss bereits mit dem Windows Kommando *mode* vorparametriert sein (Grundeinstellung). Eine Überparametrierung durch das ORG wird weitgehend durchgeführt.

Protokollparameter in der Form:

p1 **abcd**

a	0 = POA 50 wird verwendet	
	1 = POA 10 wird verwendet	
b	0 = Priorität niedrig	
	1 = Priorität hoch	
c	0 = ORG Timeout = 2s	(warten auf E/A Befehl
	1 = ORG Timeout = 30s	nach Eingabeaufforderung)
d	0 = ohne BCC	
	1 = mit BCC	

p2 = wert (ms) Timeout - warten auf ZE-Zeichenempfang (default 200ms)
p3 = wert (ms) Verzögerung Ausgabe in ms (default 0ms)
p4 = wert (ms) Verzögerung Eingabe in ms (default 0ms)

Beispiel

```
DEVICE=4,1,3964,"\\.\COM11",0111,300,100,100 ;
```
Betrieb an COM11, POA 50/ Priorität hoch/ORG Timeout 30 Sekunden/mit BCC
p2-Timeout 300ms, p3-Verz. Ausgabe 100ms, p4-Verz. Eingabe 100ms

```
DEVICE=4,2,3964,"\\.\COM12",1111 ;
```
Betrieb an COM12, POA 10/ Priorität hoch/ORG Timeout 30 Sekunden/mit BCC
sonst default

© IPKS GmbH 2007

Rechnerkopplung **DUST3964R** über eine beliebige serielle Schnittstelle

DEVICE=*anr*,gnr,3964R,"*pfadname*",*par[,t_ze[,verzaus[,verzein]]]*

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DUST3964R über eine beliebige serielle Schnittstelle definiert. Diese serielle Schnittstelle muss als COMxx unter Windows ansprechbar sein.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pfadname Pfadname der entsprechenden Windows-Geräteschnittstelle, d.h. "COMx" bzw. "\\.\COMxx". Die Schnittstelle muss bereits mit dem Windows Kommando *mode* vorparametriert sein (Grundeinstellung). Eine Überparametrierung durch das ORG wird weitgehend durchgeführt.

par Protokollparameter in der Form:

tpb

t	0 = BFA Überwachungszeit = 2 sec 1 = BFA Überwachungszeit = 30 sec
p	0 = Priorität niedrig 1 = Priorität hoch
b	0 = ohne BCC 1 = mit BCC

t_ze maximale Reaktionszeit (in ms) des ORG auf ' BFA Eingabe' (wird bei Eintrag = 0 durch Defaultwert 200ms ersetzt)

verzaus Verzögerungszeit (in ms) für den Auftrag ' BFA Ausgabe' (bei langsamen Partnersystemen, default = 0)

verzein Verzögerungszeit (in ms) für den Auftrag ' BFA Eingabe' (bei schnellen Emulator-PC's, default = 0)

Beispiel

```
DEVICE=4,1,3964R,"\\.\COM21",011
```
Betrieb an Port 21, BFA-Überwachung 2 Sekunden, Priorität hoch, mit BCC

```
DEVICE=4,2,3964R,"\\.\COM22",100
```
Betrieb an Port 22, BFA-Überwachung 30 Sekunden, Priorität niedrig, ohne BCC

```
DEVICE=4,3,3964R,"\\.\COM23",110,500,200,100
```
Betrieb an Port 23, BFA-Überwachung 30 Sekunden, mit hoher Priorität, ohne BCC, max. Reaktionszeit 500ms, Verzögerungszeit Ausgabe 200ms, Verzögerungszeit Eingabe 100ms.

© IPKS GmbH 2007

Rechnerkopplung **DUST3962** Gateway-Funktion

DEVICE=*anr*,0,3962,"SOCKET","*host*",*port*,*timeout*

Mit der Gateway-Funktionalität zur DUST 3962 (Lichtwellenleiter-Kopplung) lassen sich zwei EmulatorSysteme verbinden.

anr	Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)
host	Name des angeschlossenen Partnersystems beim aktiven Verbindungspartner Bleibt leer beim passiven Verbindungspartner
port	Portnummer/Socketnummer im eigenen Rechner (z.B.: 22220)
timeout	Überwachungszeit

Beispiel

```
DEVICE=3,0,3962,"SOCKET","system1",22220,0;
```
aktive Seite beim Verbindungsaufbau, Hostname eingetragen

```
DEVICE=3,0,3962,"SOCKET","",22221,0;
```
passive Seite beim Verbindungsaufbau, Leerstring

Rechnerkopplung **DUST3964R** über TCP/IP

DEVICE = *anr*,*gnr*,3964R,"SOCKET",port-own,
host-remote:port-remote

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DUST3964R über TCP/IP und LAN oder RAS definiert. In Windows muss ein LAN/RAS Netzwerk-Adapter und TCP/IP installiert sein (siehe auch 6.9.3).

anr	Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)
gnr	Gerätenummer gemäß ORG-Generierung (Dezimalzahl)
port-own	Portnummer/Socketnummer im eigenen Rechner (z.B.: 22223).
host-remote	Host Name oder Host Adresse im entfernten Rechner (z.B.: winnt7 oder 220.250.1.155).
port-remote	Portnummer/Socketnummer im entfernten Rechner (z.B.: 22225).

Beispiel

```
DEVICE=1,1,3964R,"SOCKET",22223,winnt7:22225;
DEVICE=1,1,3964R,"SOCKET",22223,220.250.1.155:22225;
```


© IPKS GmbH 2007

Rechnerkopplung **DUST3965R** über DF32/DF42 oder TCP/IP

DEVICE = *anr,gnr,*3965R,"*driver*",*kenn1,kenn2,size,options*

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DUST3965R über eine Schnittstelle der DF32/DF42 definiert. Die DF32/DF42 muss mit einem Protokolltreiber MSV2 installiert sein. In diesem Fall wickelt die DF32/DF42 das MSV2 Protokoll ab. Dementsprechend gilt die Beschreibung und Parametrierung dieser Baugruppe.
Alternativ ist es möglich, die DUST3965R Kopplung auf eine TCP/IP Kopplung über die Win-Socket Schnittstelle umzulenken. In diesem Fall wird eine standardmäßige LAN-Kopplung verwendet.

Wichtig!

Vor dem Start des Emulators müssen mit Hilfe des COMSOFT Programms folgende Schritte erfolgen:

- Installation des Betriebssystems auf der Baugruppe
- Installation des Protokolltreibers in der Schnittstelle
- Parametrieren des Protokolltreibers

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

driver Wahl des Gerätetreibers bzw. der Betriebsart und zwar:

"\\.\DFXX0"	für COMSOFT DF32/DF42
"SOCKET"	für TCP/IP-Übertragung über Win-Socket

kenn1 In Abhängigkeit von der Betriebsart *driver* hat diese Kennung unterschiedliche Bedeutungen:

bei DFXX0:	Boardnummer DF32/DF42 gemäß COMSOFT Beschreibung.
bei SOCKET:	"hostname" oder "Leerstring"

kenn2 In Abhängigkeit von der Betriebsart *driver* hat diese Kennung unterschiedliche Bedeutungen:

bei DFXX0:	Schnittstellen-Nr. DF32/DF42 gemäß COMSOFT Beschreibung.
bei SOCKET:	Portnummer (10000...)

size Transferpuffergröße (wie bei COMSOFT Treiberinstallation)

options Optionsfeld (derzeit keine Bits verwendet)

Beispiele

```
device=1,1,3965R,"\\.\DFXX0",0,0,512;
device=1,1,3965R,"SOCKET","WINNT1",22220,512;
```

Rechnerkopplung **DUST3966** über DF32/DF42 oder TCP/IP

DEVICE = *anr,gnr,*3966,"*driver*",*kenn1,kenn2,type,0,frame,t1,t2,t3,parfile*

Mit diesem Parameter wird die Emulation einer Rechnerkopplung DUST 3966 über eine Schnittstelle der DF32 bzw. DF42 definiert. Die DF32 bzw. DF42 muss mit einem Protokolltreiber HDLC UNC-2 oder HDLC BAC 2,8 installiert sein. In diesem Fall wickelt die DF32/42 das HDLC-Protokoll ab. Dementsprechend gilt die Beschreibung und Parametrierung dieser Baugruppe.
Alternativ ist es möglich, die DUST 3966 Kopplung auf eine TCP/IP Kopplung über die Win-Socket Schnittstelle umzulenken. In diesem Fall wird eine standardmäßige LAN-Kopplung verwendet.
Neben diesem Geräteeintrag in der Konfigurationsdatei *rpar.sys* erfordert die DUST 3966-Emulation eine Parameterdatei *parfile*, in der weitere Betriebsarteneinstellungen definiert werden. Siehe dazu auch die Beschreibung in Kapitel 6.9.

anr,gnr wie vorher

driver Wahl des Gerätetreibers bzw. der Betriebsart und zwar:

"\\.\DFXX0"	für COMSOFT DFxx
"SOCKET"	für TCP/IP-Übertragung üb. Win-Socket

kenn1 In Abhängigkeit von der Betriebsart *driver* hat diese Kennung unterschiedliche Bedeutungen:

DFXX0: Boardnummer DF32/DF42 gemäß COMSOFT Beschreibung.
SOCKET: "hostname" oder "Leerstring"

kenn2 In Abhängigkeit von der Betriebsart *driver* hat diese Kennung unterschiedliche Bedeutungen:

DFXX0: Schnittstellen-Nr. DF32/42 gemäß COMSOFT Beschreibung.
SOCKET: Portnummer (10000...)

type

B:	balanced
UP:	unbalanced primary
US:	unbalanced secondary

frame In Abhängigkeit von der Betriebsart *driver* hat dieser Parameter unterschiedliche Bedeutungen:

bei DFXX0:	0
bei SOCKET:	Maximale FRAME-Länge

t1

0:	Keine Eingabeüberwachung
1:	Eingabeüberwachung 1 Sekunde (Meldung)
5:	Eingabeüberwachung 5 Sekunden (Meldung)
24:	Eingabeüberwachung 24 Sekunden (Meldung)
101:	Eingabeüberwachung 1 Sekunde (Meldung/Verbindung trennen)
105:	Eingabeüberwachung 5 Sekunden (Meldung/Verbindung trennen)
124:	Eingabeüberwachung 24 Sekunden (Meldung/Verbindung trennen)

t2 0: Wählverbindungen in Übertragungspausen nicht trennen
1: Wählverbindungen in Übertragungspausen trennen

t3 Quittungsüberwachungszeit in Millisekunden

parfile Bei DFXX0 Pfad- und Dateiname einer Parameterdatei gemäß 5.3

 © IPKS GmbH 2007

Hinweise

- Derzeit sind die Timer t1, t2 und t3 mit 0 zu besetzen !
- Die Angabe *type* ist für den Socket-Betrieb irrelevant, muss aber angegeben werden.
- TCP/IP nur für Punkt zu Punkt-Verbindung

Beispiel

```
device=1,1,3966,"\\.\DFXX0",0,0,US,0,0,0,0,
g:\sicomp\3966us.par;
device=1,1,3966,"SOCKET","WINNT7",22227,US,1024,0,0,0;
```

Kopplung **DUST 3961** GatewayFunktion

DEVICE=*anr,gnr,*3961,"SOCKET",*ownhost,[shost:]port1[/port2],[rhost:]port1[/port2],frame,timeout*

Mit der Gateway-Funktion zur DUST 3961 lassen sich zwei EmulatorSysteme verbinden.

anr,gnr	wie oben
ownhost	eigene IP-Adresse oder HostName(*alias*) aus *...\system32\...\hosts zur* Identifikation der lokalen Netzwerkkarte. Bei nur einer NetCard im PC kann hier "." angegeben werden. Ausführliche Beschreibung, zu allen Parametern, siehe: Kapitel 6.8
shost	HostName für die TCP/IP Verbindung ' Senden '
port1	Portnummer 1 für die TCP/IP Verbindung ' Senden '
port2	Portnummer 2 für die TCP/IP Verbindung ' Senden '
rhost	HostName für die TCP/IP Verbindung ' Empfangen '
port1	Portnummer 1 für die TCP/IP Verbindung ' Empfangen '
port2	Portnummer 2 für die TCP/IP Verbindung ' Empfangen '
frame	Maximale Daten(transfer)größe in Byte
timeout	Überwachungszeit(der Verbindung) in Sekunden

Beispiel

```
device=8,0,3961,"SOCKET",PC-SICOMP,SICDEMO1:22227,
          SICDEMO2:22229,1024,10;
```

5.1.5 Sonstige Geräte

Promea-**Zeitgeber**

DEVICE = *anr,gnr,*ZIG1[,"*timeopt*"]
DEVICE = *anr,gnr,*ZIG1X[,"*timeopt*"]
Dieser Parameter definiert einen Promea-Zeitgeber.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

ZIG1 Standard (erster Aufruf *Datum/Uhrzeit einstellen* wird verworfen)
ZIG1X erster Aufruf *Datum/Uhrzeit einstellen* wird ausgeführt

timeopt Wenn man als Option NT_TIME einträgt, wird bei jedem Setzen des ORGDatums/der ORGUhrzeit auch das Windows Datum/die Windows Uhrzeit gesetzt. Wird SYS_TIME eingetragen, werden Datum/Uhrzeit unabhängig von Zeitzone und Sommer-/Winterzeit als absolute UTC Time (koordinierte Weltzeit, früher GMT) vom ORG interpretiert.

Anmerkung Wenn das ORG Datum/Uhrzeit liest, liest der Emulator Datum/Uhrzeit von Windows. Wenn über das ORG Datum/Uhrzeit verändert werden, verändert der Emulator Datum/Uhrzeit von Windows nur dann, wenn der Parameter NT_TIME angegeben ist. Sollen ORG - und Windows Datum/Uhrzeit synchron laufen, muss der Parameter NT_TIME angegeben werden, z.B. bei Betrieb einer Funkuhr. Eine Uhrzeitverstellung, z.B. Sommer/Winterzeit, muss grundsätzlich vom ORG initiiert werden (serielle Funkuhr an PROMEA oder manuelle Zeitverstellung).
☐ Uhr automatisch auf Sommer/Winterzeit umstellen - muss grundsätzlich deaktiviert sein.

Beispiel

```
DEVICE=1,1,ZIG1;
DEVICE=1,1,ZIG1,"$NT_TIME$";
DEVICE=1,1,ZIG1X,"$SYS_TIME$";
```

Zeitgeber **3691A**

DEVICE = *anr,gnr,*ALE1

Dieser Parameter definiert einen Zeitgeber 3691A.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

Beispiel

```
DEVICE=1,3,ALE1;
```

Hinweis Wird der Zeitgeber 3691A in einem SICOMP-R System ohne weitere Prozessperipherie verwendet, dann muss der Gerätename ALE1 durch ALE1ONLY ersetzt werden.

Beispiel

```
DEVICE=1,3,ALE1ONLY;
```

Hinweis Ist in einem emulierten ORG der Zeitgeber 3691-A als einziger Zeitgeber generiert und besitzt dieser keine Batteriepufferung, dann kann er ohne Änderung am emulierten ORG zum gepufferten Zeitgeber umfunktioniert werden.
Als DEVICE=... muss statt ALE1ONLY dann ALE1ONLYSP eingetragen werden.

Beispiel

```
DEVICE=1,3,ALE1ONLYSP;
```

 © IPKS GmbH 2007

Pseudogerät für **PSD**

DEVICE = *anr,gnr,* FTNT*,* "*pipename", length[,buffersize,orgerror]*

Mit diesem Parameter wird ein Pseudogerät für PSD definiert (Pipe zu SICOMP Daten). Im SICOMP R System muss dafür ein Sichtgerät DSSE/DSSA generiert sein. M2000 (*sir_ftnt.exe*) richtet dazu für jede Bearbeitungsrichtung eine *named pipe* unter Windows ein und arbeitet bezüglich dieser Pipe als **Server**. Die Pipe, über die Daten gelesen werden, wird mit der Endung **in** ergänzt, die Pipe, über die Daten geschrieben werden, wird mit der Endung **out** ergänzt.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pipename Als Pipename sind bei Benutzung der *m2psapi.dll* nur die Namen "ftnt0" bis "ftnt9“ möglich (siehe auch: PSD Dokumentation). M2000 richtet für jedes Pseudogerät zwei Pipes ein, wobei der Pipename jeweils mit den Endungen **in** und **out** ergänzt wird. Damit steht für jede Richtung eine Pipe zur Verfügung. Wird ohne PSD-Aufrufe gearbeitet, dann gibt es keine Begrenzung bei der PipeNummern Vergabe.

length Länge der Pipes in KBytes. Die Angabe 0 wird durch den Defaultwert von 4 KB ersetzt.

buffersize Größe des Eingabepuffers im Anwenderprogramm. Wenn *buffersize* angegeben und (größer) > 0 ist, dann wird der vom Anwenderprogramm zur Verfügung gestellte Eingabepuffer nach dem Empfang von Daten bis zur Länge *buffersize* aufgefüllt und mit ETX abgeschlossen.

orgerror Angegeben und = 0 : Ausgabeaufrufe des Anwenderprogramms werden immer ohne Anzeigen, unabhängig von aufgetretenen Transferfehlern, abgeschlossen.
Angegeben und ' 1 ' : Anzeigenbearbeitung wie üblich.

DEVICE=*anr,gnr,* FTNTX*,* "*pipename", length[,buffersize,orgerror]*

Mit diesem Parameter wird ein Pseudogerät für PSD definiert (Pipe zu SICOMP Daten). Im SICOMP R System muss dafür ein Sichtgerät DSSE/DSSA generiert sein. M2000 (*sir_ftnt.exe*) greift hier auf eine vorhandene *named pipe* unter Windows zu und arbeitet bezüglich dieser Pipe als **Client**. Die Pipe, über die Daten gelesen werden, wird mit der Endung **out** ergänzt, die Pipe, über die Daten geschrieben werden, wird mit der Endung **in** ergänzt.

anr,gnr wie zuvor.

pipename wie zuvor, zusätzlich mit Angabe des entfernten Rechners. Es wird keine Pipe eingerichtet.

length wie zuvor.
buffersize wie zuvor.
orgerror wie zuvor.

Wichtig!

Standard TimeOut bei "Überwachung Anruf Eingabe"
(*default 10 sec*)

Anmerkung Über Remote - Pipes können zwei SICOMP Systeme unter M2000 direkt miteinander Daten austauschen. Auf Rechner 1 wird FTNT parametriert und auf Rechner 2 wird FTNTX parametriert. Es können so mit "STAUAL Pseudogerät" auf Rechner 1 Daten ausgegeben werden, die mit "STEIAL Pseudogerät" auf Rechner 2 eingelesen werden und umgekehrt. Die Übertragung über das Rechnernetz übernimmt die Windows Funktion Remote - Pipe.
Die als Pseudogeräte verwendeten DSS-Anschlüsse sollten unter ORG-PV mit automatischer Minusquittierung generiert werden. Änderungen im generierten System können mit *TESTS* durchgeführt werden. Diesen *Eingriff ins ORG* sollte aber nur ein *Systemkenner* machen.

Beispiele
```
DEVICE=2,0,FTNT,"ftnt0",0,1024,0  ;DSSE8 Pseudogerät für PSD;
DEVICE=2,3,FTNTX,"\\server1\pipe\ftnt0",8,1024,0
                                   ;DSSE3 Pseudogerät für PSD;
```

Pseudogerät als **PipeServer/Client**

DEVICE=*anr,gnr*,FTNTD,"*pipename*",*length[,buffersize,orgerror]*
DEVICE=*anr,gnr*,FTNTDX,"*pipename*",*length[,buffersize,orgerror]*

FTNTD richtet ein Pipe-Paar ein und bearbeitet das jeweilige Pseudogerät als **Pipe-Server**.
FTNTDX stellt Verbindungen zu einem Pipe-Paar her und bearbeitet das jeweilige Pseudogerät als **Pipe-Client**.

Beide erwarten kein Längenwort vor den eigentlichen Anwenderdaten, sondern sie stellen das Längenwort beim Senden den Anwenderdaten voran und entfernen es beim Empfang (siehe Anmerkung).

anr,gnr	wie zuvor.
pipename	wie zuvor.
length	wie zuvor.
Buffersize	wie zuvor.
orgerror	wie zuvor.

Wichtig! Standard TimeOut bei "Überwachung Anruf Eingabe"
(*default 10 sec*)

Anmerkung Um bereits existierender SICOMP-Software die Möglichkeit zu geben, auch mit ''Pseudogeräten'' (statt mit realen Geräten) zu verkehren, wurde PSD durch zwei Pseudogerät-Parameter, FTNTD und FTNTDX, ergänzt. Die Konvention ''Längenwort'' (siehe unten) muss bei Einsatz dieser beiden Parameter nicht beachtet werden (siehe oben Parameterbeschreibung).
Anwender, die nicht mit dem Standardprogramm *PSDXXX* arbeiten, müssen bei der Programmierung sowohl unter Windows, als auch unter ORG-M, ORG-PV und ORG-HV eine Konvention beachten: den eigentlichen Daten muss ein Wort (SICOMP-Format) vorangestellt werden, das die Gesamtlänge der Daten in Worten beinhaltet (Längenwort). Diese Konvention hat sich aus der Telegrammschnittstelle zwischen *PSDXXX* und PSAPI entwickelt.

Beispiele
```
DEVICE=2,0,FTNTD,"ftntd0",0,512,0 ;DSSK8 Pseudogerät für PSD
DEVICE=2,3,FTNTDX,"\\server1\pipe\ftntd0",8,512,0
                                  ;DSSK3 Pseudogerät für PSD
```

 © IPKS GmbH 2007

Pseudogerät für **WINCC-PSD**

DEVICE = *anr,gnr*,WINCC,"*pipename",length[,buffersize,orgerror]*

Mit diesem Parameter wird ein Pseudogerät für WINCC-PSD definiert. Im SICOMP R System muss dafür ein Sichtgerät DSSE/DSSA generiert sein. M2000 richtet dazu für jede Bearbeitungsrichtung eine *named pipe* unter Windows ein und arbeitet bezüglich dieser Pipe als Server. Die Pipe, über die Daten gelesen werden, wird mit der Endung **in** ergänzt, die Pipe, über die Daten geschrieben werden, wird mit der Endung **out** ergänzt.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pipename Als Pipename sind beliebige Namen möglich, M2000 richtet für jedes Pseudogerät zwei Pipes ein, wobei der Pipename jeweils mit den Endungen **in** und **out** ergänzt wird. Damit steht für jede Richtung eine Pipe zur Verfügung.

length Länge der Pipes in KByte. Die Angabe 0 wird durch den Defaultwert von 4 KB ersetzt.

buffersize Größe des Eingabepuffers im Anwenderprogramm. Wenn *buffersize* angegeben und (größer) > 0 ist, dann wird der vom Anwenderprogramm zur Verfügung gestellte Eingabepuffer nach dem Empfang von Daten bis zur Länge *buffersize* aufgefüllt und mit ETX abgeschlossen..

orgerror Angegeben und = 0 : Ausgabeaufrufe des Anwenderprogramms werden immer ohne Anzeigen, unabhängig von aufgetretenen Transferfehlern, abgeschlossen.
Angegeben und ' 1 ' : Anzeigenbearbeitung wie üblich.

Beispiel

```
DEVICE=2,0,WINCC,"wincc0",8,1024,0 ; Pseudogerät für PSD
```

Anmerkung

Wenn von *PSDXXX* oder *WCCXXX* ein Schreibauftrag für den symbolischen CB-Namen "@@@@@@" oder die Objektnummer "9999" empfangen wird, dann wird mit den zugehörigen Parametern "Adresse" und "Wert" ein Anwenderunterprogramm *UPAWP* aufgerufen, das zu *PSDXXX* oder *WCCXXX* gebunden werden muss. Schreibaufträge müssen jeweils für ein einzelnes Wort abgesetzt werden, Leseaufträge sind unzulässig.

Für Anwender von WinCC und WinCC-PSD-Kanal-DLL bedeutet dies, daß man beispielsweise eine externe PSD-Variable des Typs 'A' (vorzeichenloser 16-Bit Wert) definieren kann, der der CB "@@@@@@" und die Adresse "0"(=V128) zugeordnet werden. Wenn man in diese Variable den Wert "100" schreibt (WinCC-Funktion "SetTagWord"), dann wird unter ORG-M oder ORG-PV das Unterprogramm *UPAWP* mit G6:=0 und G7:=100 aufgerufen. Damit können Anwender aus WinCC heraus beliebige (selbstprogrammierte) Funktionen unter dem emulierten ORG aktivieren. Als Kriterien dienen die V-Adresse in "Pseudo-CB" 9999 und der übergebene Wert.
PSD-Variablen, die dem CB "@@@@@@" zugeordnet werden, dürfen von WinCC nicht aktualisiert werden!
Die Standardelemente *PSDXXX* und *WCCXXX* enthalten ein Dummy-Modul. Dieses Dummy-Modul wird auf der virtuellen Platte PLSK.PSD als Element *PLSKx-PSD.PSDAWP* mitgeliefert und kann dem Anwender als Muster dienen. Es gibt lediglich die übergebenen Werte zu G6 und G7 auf dem Standardmeldegerät aus.

© IPKS GmbH 2007

Beispiel

Über WinCC-Buttons soll das ORG mit oder ohne Startliste gebootet werden können. Der PSD-Variablen "ORGBOOT" (Typ "vorzeichenloser 16-BitWert") werden CB "@@@@@@" und Adresse "3" (=V130) zugeordnet. Ein Button setzt den Wert 1(SetTagWord (ORGBOOT,1)), ein anderer Button den Wert 2 (SetTagWord (ORGBOOT,2)). In *UPAWP* wird zunächst geprüft, ob es sich um die Funktion "ORG booten" handelt, indem G6 mit 3 verglichen wird. Anschließend wird über G7 (1 oder 2) festgestellt, ob mit oder ohne Startliste gebootet werden soll, danach wird die Funktion entsprechend ausgeführt($RUFORG"...).

Hinweis

Wenn die alten Binderkarten mit den neuen Grundsprache-Elementen verwendet werden, kommt es beim Laden von *PSDXXX* und *WCCXXX* zur Fehlermeldung: "/NDEF:UPAWP". Diese Meldung kann ignoriert werden, solange die neue Funktion nicht benutzt werden soll.

Dummy-Modul

```
/#PSDAWPXX                          170800/PJG
************************************************************************
***         PSDXXX:   ANWENDERFUNKTION
************************************************************************
            'PKEN' ,,,,0
            'DEEX' UPAWP
            'NDEX' STAMELD

            'NAME' PSDAWP
            'SATZ' PSDAWP

'VA' TEXTA/
' Y'         = 'Z=PSDAWP: DUMMY: #### ####(3)'
'VA' TEXTE/

'IA' UPAWP/    ('R2) := R7          *** DUMMY!
' A'         ('R2) := R3          ***
' A'         ('R2) := R4          ***
' A'         ('R2) := R5          ***

' A'         *** G6               *** RELATIVE ADRESSE (AB 0)
' A'         *** G7               *** AKTUELLER WERT

' A'         R3 := <7+TEXTA>              ***
' A'         R4 := 'H=8000'               ***
' A'         R5 := G6                     ***
' A'         $BIHEX;                      ***                  N
' A'         'IA'                                              B
            ='H=E000'                                         B
             ***                                              B
' A'         R3 := <10+TEXTA>             ***
' A'         R4 := 'H=0000'               ***
' A'         R5 := G7                     ***
' A'         $BIHEX;                      ***                  N
' A'         'IA'                                              B
            ='H=E000'                                         B
             ***                                              B
' A'         R5 := (R2')          ***
' A'         R4 := (R2')          ***
' A'         R3 := (R2')          ***

' A'         :SP (R2')            ***

'ENDE'
```


© IPKS GmbH 2007

SonderPromea **MX1**

DEVICE = *anr,gnr,*EAMX1,"*pfadname*",*time,bytes,flags*

EAMX1 bearbeitet ausschließlich Eingaben über einen seriellen Port an das ORG. Kriterium für das Eingabeende ist entweder der Ablauf einer Überwachungszeit nach Empfang des jeweils letzten Zeichens oder das Erreichen einer maximalen Grenze. Per Parameter kann weiterhin ausgewählt werden, ob die empfangenen Daten um ETX ergänzt werden sollen.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)

gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

pfadname Bezeichnung der unter Windows definierten seriellen Schnittstelle. Der *pfadname* wird in der Form "COMx" bzw. "\\.\COMxx" notiert.

time Überwachungszeit in Millisekunden. Trifft nach dem Empfang des letzten Zeichens innerhalb dieser Zeit kein weiteres Zeichen ein, werden die gesammelten Daten an das ORG übergeben. Die Angabe "0" schaltet die Zeitüberwachung ab.

bytes Maximale Zeichenanzahl. Sind soviele Zeichen empfangen worden, wie diese Grenze vorgibt, werden die gesammelten Daten an das ORG übergeben. Die Angabe "0" schaltet die Zählüberwachung ab.

flags Parametrierung
- Bit 0 = 0 Daten unverändert an das ORG übergeben.
- Bit 0 = 1 Daten mit ETX ergänzt an das ORG übergeben.
- Bit 1 = 0 Daten als Bytestream an das ORG übergeben.
- Bit 1 = 1 Daten als Wortstream an das ORG übergeben.
- Bit 3 = 0 Seriellen Port wie üblich parametrieren.
- Bit 3 = 1 Seriellen Port nicht parametrieren, MODE-Kommando ist gültig.
- Bit 4 = 0 Zeichen unverändert übernehmen und an das ORG übergeben.
- Bit 4 = 1 nur die 7 niederwertigen Bit an das ORG übergeben.

Beispiel

```
DEVICE=3,0,EAMX1,"\\.\COM13",70,20,0x9;
```

Eingaben über seriellen Port 13; Überwachungszeit 70 ms;
nach 20 Byte Transfer des Puffers, ergänzt durch ETX, an das ORG; serieller Port wird über MODE-Kommando parametriert.
(0x9 = Bit 0 und Bit 3 gesetzt)

© IPKS GmbH 2007

BDE-Terminal **ES**

DEVICE = *anr,gnr*,EAES,"*pfadname*",*options*

Dieser Parameter definiert den Anschluss eines BDE-Terminals der Terminalfamilie ES 1 über serielle Schnittstelle.

Hinweis Die Parametrierung des seriellen Ports mit dem MODE-Kommando (*setall.bat*) muss unbedingt vor dem Emulator-Start erfolgt sein.

anr Anschlussstellennummer des Gerätes gemäß ORG-Generierung (Dezimalzahl)
gnr Gerätenummer gemäß ORG-Generierung (Dezimalzahl)

EAES Gerätebezeichnung

pfadname Bezeichnung der unter Windows definierten seriellen Schnittstelle. Der Pfadname wird in der Form "COMx" bzw. "\\.\COMxx" notiert.

options Parametrierung
Bit 4 = 0 Eingabe nicht transparent
Bit 4 = 1 Eingabe transparent
Bit 5 = 0 Ausgabe nicht transparent
Bit 5 = 1 Ausgabe transparent
Bit 6 = 0 Betrieb über Laufprozedur
Bit 6 = 1 Betrieb über gesicherte 38xx-Prozedur
Bit 7 = 0 muss 'Null' sein, sonst IPKS-Testmodus aktiv

Beispiel
```
DEVICE = 1,1,EAES,"\\.\COM2",0;
```
Ein- und Ausgabe nicht transparent, Betrieb über Laufprozedur.

```
DEVICE = 1,1,EAES,"\\.\COM2",0x40;
```
Ein- und Ausgabe nicht transparent, Betrieb über gesicherte 38xx-Prozedur.

```
DEVICE = 1,1,EAES,"\\.\COM2",0x70;
```
Ein- und Ausgabe transparent, Betrieb über gesicherte 38xx-Prozedur.

Parameter **EAU**

EAU

Wird dieser Parameter angegeben, werden Befehle an EA-U Baugruppen nicht leer durchlaufen, sondern an den jeweiligen Geräteprozess weitergegeben.

Emulation **KS N16**

DEVICE = *anr,*0,CS275,"\\.\NATDEV",*zyklus*

Dieser Parameter definiert die Anbindung an TELEPERM M Systeme.
(Anbindung an einen CS275-Bus über die Baugruppe KS N16)

Hinweis

Die KS N16 hat immer zwei Schnittstellen zum ORG; deshalb wird statt der Gerätenummer ' gnr ' der Wert ' 0 ' angegeben. Der Prozess SIR_CS27.EXE bearbeitet als Ausgabegerät immer Gerät 0, als Eingabegerät immer Gerät 1 der jeweils parametrierten AnschlussstellenNummer.

anr Anschlussstellennummer, unter der die KS N16 im Originalsystem generiert ist. (Dezimalzahl)
zyklus Lesezyklus in Millisekunden.
(Bei Angabe von 0 wird ein Defaultzyklus von 10 ms eingestellt)

Beispiel
```
DEVICE = 7,0,CS275,"\\.\NATDEV",300;
```


© IPKS GmbH 2007

6 Details zu den Geräteemulationen

6.1 Geräteemulation DS074

6.1.1 Bildschirmaufbau

Die Emulation der Datensichtstation DS074 erfolgt in einem Windows Fenster als Konsolen-Applikation.

Dabei sind folgende Bedingungen gegeben:

- Der unter *pipename* parametrierte Name erscheint als Fensterüberschrift.
- In der Fensterdarstellung (default) wird mit einem Zeichensatz 9x15 gearbeitet (Standard Window Screen).
- Vollbildmodus kann über: **re. Maustaste**–OK–Screen Parameter–*Screen Size* eingestellt werden (Full Screen).
- Der Bildschirmbereich im Fenster ist unterteilt in:
  den Datenbereich (Zeile 1 bis 24) und in die Statuszeile (Zeile 25). Die Statuszeile dient sowohl zur Darstellung von Tabulatoren als auch von aktuellen Betriebszuständen (Rollmode, Block-/Zeichenbetrieb, virtuelle Konsole).

Zur Zeit sind folgende Betriebszustände realisiert:

Anzeige in Statuszeile	Roll	Rollbetrieb Seitenbetrieb	Shift+F6
	Block Zeich	Blockbetrieb Zeichenbetrieb	Shift+F7
Nur SIC-M	VK	virtuelle Konsole ein Normalbetrieb	Shift+F8

Die Betriebszustände können mit Shift+F6, Shift+F7 und Shift+F8 geändert werden.

Umschalten auf andere Terminalseiten (nur TE_ALPHA)	Shift + Pfeiltaste nach oben	nächste Bildschirmseite
	Shift + Pfeiltaste nach unten	vorherige Bildschirmseite

Die Nummer (Ziffer 2-6) der aktuellen Bildschirmseite wird in der Statuszeile der Terminalemulation angezeigt.
Detaillierte Informationen zu allen Funktionen können dem Benutzerhandbuch Terminalemulation entnommen werden.

© IPKS GmbH 2007

6.1.2 Tastenbelegung

Enter	DÜZ
End	DÜ
Ctrl + End	DÜM
Ctrl + B	STX
Ctrl + C	ETX
Ctrl + G	BEL
Ctrl + L	FF (Bildschirm löschen, Cursor auf Bildschirmanfang)
Ctrl + U	Sichtgerät rücksetzen (Block- und Maskenbetrieb)
Backspace	Cursor links
Tab	Tabulatorsprung rechts
Shift + Tab	Tabulatorsprung links
Ctrl + Tab	Tabulatormarke setzen
Ctrl + Shift + Tab	Tabulatormarke rücksetzen
Ins	Zeichen einfügen
Del	Zeichen ausfügen
+	Zeile einfügen (Plus-Taste im Numerikblock)
-	Zeile ausfügen (Minus-Taste im Numerikblock)
Esc	KT '0'
F1 bis F9	KT '1' bis KT '9'
F10	KT ':'
F11	KT ';'
F12	KT '<'

© IPKS GmbH 2007

Alt 0	KT '0'
bis	
Alt 9	KT '9'
Alt A	KT 'a'
bis	
Alt Z	KT 'z'
Alt .	KT ' . '
Alt Shift A	KT 'A'
bis	
Alt Shift Z	KT 'Z'

Betriebsartenumschaltung:

Shift F6	Ein-/Ausschalten Rollbetrieb
Shift F7	Umschaltung Block-/Zeichenbetrieb
Shift F8	Ein-/Ausschalten Virtuelle Konsole

Bildschirmseitenumschaltung:

Shift ↑	nächste Bildschirmseite
Shift ↓	vorherige Bildschirmseite

nur für Zeichenbetrieb:

End	Datenübergabe (DUE)
Ctrl Enter	Sichern ohne Abschluss (DÜZ)
Enter	LF (Cursor auf Anfang der nächsten Zeile)
Ctrl End	Abschluss mit Sichern (DÜM)
PgUp	rückwärts blättern (Ctrl R)
PgDn	vorwärts blättern (Ctrl N)

Anmerkung Die Tastenbelegung wird über die Parameterdatei *dsskx.tko* festgelegt. Die beschriebene Tastenbelegung entspricht der mit M2000 ausgelieferten Parameterdatei.

© IPKS GmbH 2007

6.2 Rechnerkopplung DU05

Die DU05 Geräteemulation bietet grundsätzlich zwei Betriebsarten:

- Emulation der DU05 über DF42
- Umlenkung der DU05 Kommunikation auf TCP/IP

6.2.1 Emulation DU05 über DF42

Die DU05 Geräteemulation bearbeitet das Übertragungsprotokoll MSV2.
Damit sind beispielsweise folgende Kopplungen realisierbar:

- M2000 - M2000 über MSV2 und SINEC
- M2000 - Fremdsysteme über MSV2

Nicht implementiert sind die WIKO-Funktionen (Remote-Laden...)

6.2.1.1 Voraussetzung

Die Betriebssoftware für die eingesetzte Hardware wird von M2000 implizit installiert.

Zusätzlich zu dem Geräteeintrag (*device = ..*) in der Konfigurationsdatei *mpar.sys* erwartet die DU05 Geräteemulation zusätzliche Parameter in einer Parameterdatei, deren Namen und Verzeichnis frei wählbar sind und im Geräteeintrag zugeordnet werden.

6.2.1.2 Parametrierung

Hinweis!

Alle über COMSOFT Baugruppen betriebenen Kopplungen müssen vor dem Emulatorstart mit dem COMSOFT Programm DFTEST parametriert werden.

Die anwendungsspezifischen Parameter werden in einer ASCII-Datei hinterlegt. Die Parameterzeilen können in Gross- und Kleinbuchstaben notiert werden, Kommentare werden mit '#' eingeleitet und gelten bis zum Zeilenende.

Folgende Parameter sind einstellbar, wobei alle Angaben zu den Parametern (einschließlich Default-Einstellungen) der Beschreibung zum jeweiligen COMSOFT-Treiber zu entnehmen sind:

PATH *pfadname*	Verzeichnis mit COMSOFT-Komponenten
BOARD *nr*	Board-Nummer
UNIT *nr*	Schnittstellen-Nummer auf dem Board
DU_MODUS *wert*	0x00
OSFILE	OS42.BIN

 © IPKS GmbH 2007

6.2.1.3 Protokollparameter für MSV2

Folgende Protokollparameter sind zum Treiber MSV2 einstellbar:

BAUDRATE	value
MODEM_SET	value (0..1)
MODEM_TO	value (1..255)
BITS	0x00
T1	value
T2	value
T3	value
T4	value
MZ1	value
MZ2	value
MZ3	value
MZ4	value
MULTI	value

6.2.2 Umlenkung der DU05 Kommunikation auf TCP/IP

In der Datei *mpar.sys* eines der beiden beteiligten Rechner wird der Host-Name des Partnerrechners eingetragen, wie er auch in der Datei *...\system32\drivers\etc\hosts* eingetragen ist; dieser Rechner verhält sich später beim Verbindungsaufbau aktiv. Auf dem anderen beteiligten Rechner wird in die Datei *mpar.sys* als Host-Name ein Leerstring eingetragen; dieser Rechner verhält sich später beim Verbindungsaufbau passiv.
Die Übertragung der Daten erfolgt verbindungsorientiert über das Level 4 Transportprotokoll TCP.
Die Portnummer muss größer/gleich 10 000 angegeben werden, sie ist auf beiden Rechnern identisch.

Spezifische Anlauffunktionen sind nicht notwendig.

6.3 Rechnerkopplung DU06

Die DU06 Geräteemulation bietet grundsätzlich zwei Betriebsarten:

- Emulation der DU06 über DF32 bzw. DF42
- Umlenkung der DU06 Kommunikation auf TCP/IP

6.3.1 Emulation DU06 über DF32/42

Die DU06 Geräteemulation bearbeitet die Übertragungsprotokolle HDLC U/P, HDLC U/S und HDLC B.

Damit sind beispielsweise folgende Kopplungen realisierbar:

- M2000 - M2000 über HDLC B oder HDLC U und SINEC
- M2000 - IBM über SDLC und SINEC SNSNA
- M2000 - X.25 Netz über HDLC B und SINEC SNPV.

Nicht implementiert sind die WIKO-Funktionen (Remote-Laden...)

6.3.1.1 Voraussetzung

Die Betriebssoftware für die eingesetzte Hardware wird von M2000 implizit installiert.

Zusätzlich zu dem Geräteeintrag (*device* = ..) in der Konfigurationsdatei *mpar.sys* (siehe 4.2.4 Kopplungen) erwartet die DU06 Geräteemulation zusätzliche Parameter in einer Parameterdatei, deren Namen und Verzeichnis frei wählbar sind und im Geräteeintrag zugeordnet werden.

Die Bearbeitung der Funktion '**SIM**' muss im DFXX-Treiber separat eingestellt werden, die Parametrierung wird vom Emulator nicht unterstützt.
Eine DFXX-Baugruppe, die vom Emulator als 'DU06 unbalanced' mit **SIM** Unterstützung verwendet werden soll, muss vor dem Start des Emulators mit dem Programm DFTEST parametriert werden.
Die DU06 Parameterdatei (siehe nächste Seite) muss die Einträge BOARD, UNIT, und PATH, sowie die Zeile 'DU_MODUS 0' enthalten. Alle übrigen Einträge werden dann ignoriert.

 © IPKS GmbH 2007

6.3.1.2 Parametrierung

Die anwendungsspezifischen Parameter werden in einer ASCII-Datei hinterlegt. Die Parameterzeilen können in Gross- und Kleinbuchstaben notiert werden, Kommentare werden mit '#' eingeleitet und gelten bis zum Zeilenende.

Folgende Parameter sind für beide Protokollarten einstellbar, wobei alle Angaben zu den Parametern (einschließlich Default-Einstellungen) der Beschreibung zum jeweiligen COMSOFT-Treiber (HDLC-UNC oder HDLC-BAC) zu entnehmen sind:

PATH *pfadname* Verzeichnis mit COMSOFT-Komponenten
BOARD *nr* Board-Nummer
UNIT *nr* Schnittstellen-Nummer auf dem Board
DU_MODUS *wert* Bitweise Einstellung von Sonderfunktionen mittels eines 32-Bit-Wertes:

- Bit 0 = 0 Die Baugruppe wird beim Start der Emulation einmalig mit Treibern versorgt und parametriert.
- Bit 0 = 1 Die Baugruppe wird bei jedem ORG-Anstoss 'Basisparametrierung' mit Treibern versorgt und parametriert.
- Bit 1 = 0 keine 3277-Verbindung zu IBM
- Bit 1 = 1 3277-Verbindung zu IBM
- Bit 2 = 0 Eine HDLC-Verbindung wird ausschließlich mit dem COMSOFT-Treiber Aufruf 'CLOSE_UNIT' abgebaut.
- Bit 2 = 1 Eine HDLC-Verbindung wird mit dem COMSOFT-Treiber-Aufruf 'CLOSE_UNIT' und anschließender Treiberparametrierung abgebaut.

```
# Beim Einsatz einer DF42 Baugruppe ist die folgende Zeile zu
# aktivieren
# OSFILE OS42.BIN
```

Beim Betrieb einer DU06B, DU06UP oder einer DU06US über eine COMSOFT-Baugruppe muss derzeit in Parameter DU_MODUS Bit 2 gesetzt werden, damit nach dem logischen Verbindungsaufbau auch die physikalische Verbindung abgebaut wird (Anpassung an aktuelles Verhalten der COMSOFT-Treiber).

© IPKS GmbH 2007

6.3.1.3 Protokollparameter für HDLC-UNC

Beim Betrieb einer DU06UP oder einer DU06US über eine COMSOFT-Baugruppe können ab COMSOFT-Treiber DHDLC.DRV C01 auch Modemsignale bearbeitet werden. Dazu sind die neuen Parameter MODEM und MODEM_TIMEOUT einzustellen. Neben dem COMSOFT-Treiber DHDLC.DRV sind auch die COMSOFT-Komponenten DHDLC.DAT, OS32.BIN und DFXX0 neu in das Windows-System einzubringen.
Dabei muss DFXX0 in das Verzeichnis *%systemroot%\system32\drivers* eingebracht werden, anschließend ist Windows neu zu starten.

Folgende Protokollparameter sind zum Treiber HDLC-UNC einstellbar:

MODEM	YES | NO	
MODEM_TIMEOUT		
SNRMREP	*value*	(0..255)
NRZ	YES | NO	
CRC	0 | 1	
MASTER	YES | NO	
CLOCK	*value*	(0..3)
MULTIPOINTYES | NO		
CREDIT	*value*	(0..7)
IFRAMESIZE	*value*	(128..1024)
STATIONS	*value*	(1..32)
HDLCADR	*adr*	
TIMER1	*value*	
TIMER2	*value*	
BAUDRATE	*value*	

6.3.1.4 Protokollparameter für HDLC-BAC

Folgende Protokollparameter sind zum Treiber HDLC-BAC einstellbar:

DCE	YES | NO	
MODEMDEF	*value*	(0..15)
MODEM_I_TO	*value*	(0..255)
MODEM_I_OK	*value*	(0..255)
AUTORESTART	YES | NO	
SENDSABM	YES | NO	
REP	*value*	(0..255)
NRZ	YES | NO	
CRC	0 | 1	
RNRBRK	*value*	(0..255)
CLOCK	*value*	(0 | 1 | 4)
CREDIT_HDLC	*value*	(0..7)
V24V11	0 | 1	
IFRAMESIZE	*value*	(64..1024)
HDLCADRCON	*adr*	
HDLCADROWN	*adr*	
TIMER1	*value*	
TIMER2	*value*	
BAUDRATE	*value*	

© IPKS GmbH 2007

6.4 Offene DU0x-Kommunikations-Schnittstelle

6.4.1 Überblick

Die ‘’offene DU0x Kommunikations-Schnittstelle’’ des Emulators M2000, (Variante SICOMP M), ermöglicht die Umleitung von Ein-/Ausgaben, die über eine im emulierten ORG-M generierte Baugruppe DU0x erfolgen, auf eine TCP/IP-Socket Verbindung. Damit können ORG-M Anwendungen Daten mit allen Systemen austauschen, die über TCP/IP erreichbar sind, und auf denen ein oder mehrere Partnerprogramme gemäß dieser Spezifikation agieren.
Über diese Schnittstelle lassen sich auch zwei emulierte ORG-M miteinander verknüpfen. Während des Verbindungsaufbaus muss ein Partner aktiv, der andere passiv agieren. Nach Aufbau der Verbindung kann jeder Partner jederzeit Daten senden und empfangen.

Hinweis Die Darstellung numerischer Werte kann auf verschiedenen Hardware-Plattformen unterschiedlich sein; bei der Definition der ausgetauschten Anwenderdaten ist dieser Sachverhalt zu berücksichtigen. Die hier beschriebene Schnittstelle leitet in beide Richtungen alle Daten unverändert weiter. Von einer Anwendung, die unter einem emulierten ORG-M abläuft, ausgegebene numerische Daten, werden grundsätzlich in SICOMP-M Darstellung übertragen. Eine der beiden beteiligten Anwendungen muss also, falls erforderlich, diese Daten konvertieren.

6.4.2 Umlenkung der DU04 Kommunikation auf TCP/IP

In der Parameterdatei der beiden beteiligten Rechner wird der Host-Name oder die Host Adresse des Partnerrechners eingetragen, wie er auch in der Datei *...\system32\drivers\etc\hosts* eingetragen ist.
Die Übertragung der Daten erfolgt verbindungslos über das Level 4 Transportprotokoll UDP. Auf Anwendungsebene sollte ein Quittungsverkehr implementiert sein (beispielsweise wie innerhalb von SINEC VS).
Diese Übertragung eignet sich auch dafür, um einfache Kopplungen zu Fremdsystemen wie UNIX zu realisieren.
Die Portadressen für jeden Rechner müssen eindeutig sein, es muss eine Zahl von mindestens 10000 angegeben werden.
Spezifische Anlauffunktionen sind nicht notwendig.

6.4.3 Umlenkung der DU05 Kommunikation auf TCP/IP

Verbindungsaufbau

Beide Verbindungspartner richten jeweils einen TCP/IP-Socket (Adressfamilie AF_INET, Typspezifikation SOCK_STREAM) ein, über den alle Daten und Quittungen ausgetauscht werden.
Auf Emulatorseite werden die Socketoptionen 'REUSEADDR' und 'KEEPALIVE' gesetzt.
Der Emulator kann über die Parametrierung beim Verbindungsaufbau als aktiver oder als passiver Partner eingestellt werden. Der passive Partner wartet mit ACCEPT auf den Verbindungsaufbau, der aktive Partner versucht den Verbindungsaufbau mit CONNECT.
Das Anwenderprogramm auf dem Partnersystem sollte entsprechend arbeiten (gleiche Optionen, sofortiger Verbindungsneuaufbau nach Programmstart oder Verbindungsabbruch, usw.).

Verbindungsabbau / Verbindungsabbruch

Der Emulator bricht die Socket-Verbindung vor dem Programmende oder nach dem Ablauf der Verbindungsüberwachung ab.

Datenverkehr **Senden von der ORG-M-Anwendung zur Partneranwendung**

Die ORG-M-Anwendung macht eine Ausgabe an die jeweilige DU05 (Aufruf KOPAUSH), dabei dürfen nur die Aufrufoptionen 'Daten', 'codegebunden / transparent' und ggfs. 'Blockende ETX / Blockende ETB' verwendet werden, nicht aber 'Header', 'Header und Daten', 'Status' usw.
Diese Sendelänge wird vom Emulator als Datenwort (INTEL-Darstellung) den eigentlichen Anwenderdaten vorangestellt und mit an das Partnersystem gesendet.
Nach dem Empfang der Daten durch das Partnersystem muss dieses eine (Transport-)Quittung zurücksenden

Daten
(byteweise Darstellung)

Länge Anwenderdaten in Bytes (Low-Byte)
Länge Anwenderdaten in Bytes (High-Byte)
Anwenderdaten... (Erstes Byte)
...
Anwenderdaten (Letztes Byte)

Positive Quittung
zum Emulator
(byteweise Darstellung)

0xFF
0xFF

Negative Quittung
zum Emulator
(byteweise Darstellung)

0xFE
0xFF

Senden von der Partneranwendung zur ORG-M-Anwendung
Die ORG-M-Anwendung wartet auf eine Eingabe über die jeweilige DU05.
(Aufruf KOPEINH)
Der DU05-Geräteprozess wertet nach dem Empfang von Daten über die TCP/IP-Verbindung die ersten beiden Bytes aus (Sendelänge oder Quittungskennung). Nach dem Empfang von Daten werden diese mit einem Begrenzungszeichen ETX abgeschlossen und **ohne** das Wort Telegrammlänge an die wartende ORG-M-Anwendung übergeben. Anschließend wird eine positive Quittung an das Partnersystem zurückgesendet.

 © IPKS GmbH 2007

6.4.4 Umlenkung der DU06 Kommunikation auf TCP/IP

In der Parameterdatei eines der beiden beteiligten Rechner wird der Host-Name des Partnerrechners eingetragen, wie er auch in der Datei *...\system32\drivers\etc\hosts* eingetragen ist; dieser Rechner verhält sich später beim Verbindungsaufbau aktiv. Auf dem anderen beteiligten Rechner wird als Host-Name ein Leerstring eingetragen; dieser Rechner verhält sich später beim Verbindungsaufbau passiv.
Die Übertragung der Daten erfolgt verbindungsorientiert über das Level 4 Transportprotokoll TCP.
Die Portadresse ist auf beiden Rechnern identisch, es muss eine Zahl von mindestens 10000 angegeben werden.

Spezifische Anlauffunktionen sind nicht notwendig.

© IPKS GmbH 2007

6.5 Kommunikationssteuerung CP1400

6.5.1 Installation

Der im folgenden beschriebene Installationsvorgang einer H1 Kopplung ist durch den Installationsablauf am Original SICOMP M System vorgegeben. Es wird davon ausgegangen, dass dieser Ablauf bekannt ist.

Auf dem PC, auf dem auch M2000 abläuft, sind eine Baugruppe CP1413 oder CP1613 und unter Windows die zugehörige Software SIMATIC NET zu installieren. Weiterhin ist unter Windows noch das Parametrierprogramm NML zu installieren, mit dem eine H1 Datenbasis erzeugt wird.
Es wird davon ausgegangen, dass im ORG-M des original SICOMP Systems das sogenannte AP-System für den Einsatz des CP1400 installiert ist.

Im ersten Schritt werden unter Windows mit dem Programm NML die CP1400 Parameterdateien erzeugt (beschrieben in der CP1400 Dokumentation). Die erzeugten Dateien haben die Namen *xyz.ADB* und *xyz.LDB*. Die Datei *xyz.ADB* ist anschließend in Datei *APDBA0,* und die Datei *xyz.LDB* ist in Datei *APDBA1* zu kopieren.

Im zweiten Schritt wird unter Windows mittels SIMATIC NET auf die Datei *xyz.LDB* zugegriffen. Hierbei erscheint eine Fehlermeldung, die quittiert werden muss. Daraufhin wird eine Datei *xyz.TXT* erzeugt. In der Datei *xyz.TXT* müssen die als fehlerhaft markierten Zeilen am Dateiende mit einem Texteditor (z.B. NOTEPAD) gelöscht werden.

Im dritten Schritt wird wiederum unter Windows über SIMATIC NET auf die neu erzeugte Datei *xyz.TXT* zugegriffen. Mit dieser Datei wird als Muster eine Binärdatei *xyz1.LDB* generiert. Hierbei sind gegebenenfalls noch Anpassungen vorzunehmen. Die Datei *xyz.LDB* wird dann als Datenbasis für den CP1413 oder CP1613 verwendet.

Im vierten Schritt wird erstmals M2000 gestartet. Das unter M2000 ablaufende ORG sollte ohne Startliste anlaufen. Weiterhin sollte vom ORG aus auf die mit M2000 ausgelieferte "Platte" PLSK.PSD zugegriffen werden können. Von der "Platte" PLSK.PSD sind aus der Bibliothek GSB die Programme PCCOPY und CP1400 im ORG zu laden (PRP oder HRP). Mit dem Programm PCCOPY werden die im ersten Schritt unter Windows erstellten Dateien *APDBA0* und *APDBA1* auf die ORG Platte kopiert, auf der sich die Datenbasen des AP-Systems befinden. Anschließend ist einmalig im ORG das neu geladene Programm CP1400 zu starten. Das Programm CP1400 verbindet die Programme des AP-Systems mit den kopierten Datenbasen *APDBA0* und *APDBA1*.
Das Programm CP1400 muss nach jedem einkopieren von neuen Datenbasen einmal ablaufen. Es kann auch bei jedem SICOMP Systemanlauf vor den Komponenten des AP-Systems gestartet werden.
Die Bedienung :APADMI:CHANGE STARTID TO ACTUAL entfällt.

Im letzten Schritt wird M2000 beendet und Windows gebootet. Nach dem Start von M2000 und dem Anlauf des unter M2000 ablaufenden SICOMP Systems arbeitet der CP1400 mit den neuen Parametern.

 © IPKS GmbH 2007

6.6 Kommunikationssteuerung CS275

6.6.1 Installation der Baugruppe

Die NAT Baugruppe ist entsprechend der mitgelieferten Dokumentation im PC zu installieren.
Für die Baugruppe muss kein Interrupt eingestellt werden. Die auf der Baugruppe eingestellte IO-Adr. sollte man sich merken (siehe *NatDev.ini*).

6.6.2 Software Installation

Dem Optionspaket M2-CS (Kommunikationssteuerung KS N16/CS275, Anbindung an TELEPERM) liegt eine ausführliche Installationsanweisung bei.

Siehe auch: Kap. 4.2.4, Kopplungen/CS275
Kap. 5.1.5, Sonstige Geräte/KS N16

© IPKS GmbH 2007

6.7 Rechnerkopplung DUST3965R

Die DUST3965R Geräteemulation bietet grundsätzlich zwei Betriebsarten:

- Emulation der DUST3965R über DF42
- Umlenkung der DUST3965R Kommunikation auf TCP/IP

6.7.1 Emulation DUST3965R über DF42

Die DUST3965R Geräteemulation bearbeitet das Übertragungsprotokoll MSV2. Damit sind beispielsweise folgende Kopplungen realisierbar:

- M2000 - M2000 über MSV2 und SINEC
- M2000 - Fremdsysteme über MSV2

Nicht implementiert sind die WIKO-Funktionen (Remote-Laden...)

6.7.1.1 Voraussetzung

Die Betriebssoftware für die eingesetzte Hardware wird von M2000 implizit installiert.

Zusätzlich zu dem Geräteeintrag (*device* = ..) in der Konfigurationsdatei *rpar.sys* erwartet die DUST3965R Geräteemulation zusätzliche Parameter in einer Parameterdatei, deren Namen und Verzeichnis frei wählbar sind und im Geräteeintrag zugeordnet werden.

6.7.1.2 Parametrierung

Hinweis!

Alle über COMSOFT Baugruppen betriebenen Kopplungen müssen vor dem Emulatorstart mit dem COMSOFT Programm DFTEST parametriert werden.

Die anwendungsspezifischen Parameter werden in einer ASCII-Datei hinterlegt. Die Parameterzeilen können in Gross- und Kleinbuchstaben notiert werden, Kommentare werden mit "#" eingeleitet und gelten bis zum Zeilenende.

Folgende Parameter sind einstellbar, wobei alle Angaben zu den Parametern (einschließlich Default-Einstellungen) der Beschreibung zum jeweiligen COMSOFT-Treiber zu entnehmen sind:

PATH *pfadname*	Verzeichnis mit COMSOFT-Komponenten
BOARD *nr*	Board-Nummer
UNIT *nr*	Schnittstellen-Nummer auf dem Board
DU_MODUS *wert*	0x00
OSFILE	OS42.BIN

 © IPKS GmbH 2007

6.7.1.3 Protokollparameter für MSV2

Folgende Protokollparameter sind zum Treiber MSV2 einstellbar:

BAUDRATE	value
MODEM_SET	value (0..1)
MODEM_TO	value (1..255)
BITS	0x00
T1	value
T2	value
T3	value
T4	value
MZ1	value
MZ2	value
MZ3	value
MZ4	value
MULTI	value

© IPKS GmbH 2007

6.8 Rechnerkopplung DUST3961, GatewayFunktion

Je Parameter(Baugruppe) 3961 werden zwei TCP/IP-Sockets (Adressfamilie AF_INET, Typspezifikation SOCK_STREAM) eingerichtet. Über den ersten Socket werden Daten an das Partnersystem gesendet, über den zweiten werden Daten vom Partnersystem empfangen.
Der Emulator setzt die Socketoptionen REUSEADDR und KEEPALIVE, die Datenfeldgröße wird entsprechend den Emulatorparametrierungen eingestellt.
Auf Emulatorseite kann je Socket eingestellt werden, ob der Emulator beim Verbindungsaufbau aktiv oder passiv sein soll. Der Emulator versucht ab EmulatorStart, die Verbindungen (aktiv oder passiv) aufzubauen. Bestehende Verbindungen bleiben aufgebaut, bis einer der beiden Partner einen Verbindungsabbau einleitet oder bis ein Verbindungsfehler auftritt.
Nach einem Verbindungsabbau oder einem Verbindungsabbruch versucht der Emulator sofort, die jeweils abgebrochenen Verbindungen wieder aufzubauen.
Alle ausgetauschten Daten enthalten in den ersten vier Bytes die Nettolänge des jeweiligen Telegramms (32-Bit-Wert in Intel-Darstellung).
Beim Senden an das Partnersystem stellt die 3961-Gatewayfunktion diesen 32-Bit-Wert den vom ORG an das Gerät 3961 übergebenen Daten voran.
Beim Empfang vom Partnersystem entfernt die 3961-Gatewayfunktion diesen 32-Bit-Wert, bevor die Daten an das ORG übergeben werden.

Device = anr,gnr,3961,"SOCKET", ownhost, sendeverbindung, empfangsverbindung, datenlaenge, timeout

ownhost

Eigene TCP/IP-Adresse oder eigener TCP/IP-Aliasname aus *...\system32\...\hosts* zur Identifikation der lokalen Netzwerkkarte.
Beim Betrieb auf einem Rechner mit nur einer Netzwerkkarte kann hier "**.**" angegeben werden.

sendeverbindung

Syntax: [*hostname:*]*port1*[/*port2*]
Über die hier parametrierte Verbindung werden Daten an den Partner gesendet.
Wird *hostname:* angegeben, dann wird die Verbindung zum Partner aktiv aufgebaut. Lokal wird der Socket *ownhost:port1* eingerichtet und die Verbindung zum Port *hostname*:*port2* aufgebaut. Ist *port2* nicht angegeben, dann wird *port2* mit *port1* gleichgesetzt.
Wird *hostname*: nicht angegeben, dann wird die Verbindung zum Partner passiv aufgebaut. Lokal wird der Socket *ownhost*:*port1* eingerichtet, anschließend wird auf Verbindungsaufbau des Partners gewartet. Eine Angabe von *port2* wird bei passivem Verbindungsaufbau nicht verwendet.

empfangsverbindung

Aufbau: [*hostname*:]*port1*[/*port2*]
Über die hier parametrierte Verbindung werden Daten vom Partner empfangen.
Wird *hostname*: angegeben, dann wird die Verbindung zum Partner aktiv aufgebaut. Lokal wird der Socket *ownhost*:*port1* eingerichtet und die Verbindung zum Port *hostname*:*port2* aufgebaut. Ist *port2* nicht angegeben, dann wird *port2* mit *port1* gleichgesetzt.
Wird *hostname*: nicht angegeben, dann wird die Verbindung zum Partner passiv aufgebaut. Lokal wird der Socket *ownhost*:*port1* eingerichtet, anschließend wird auf Verbindungsaufbau des Partners gewartet. Eine Angabe von *port2* wird bei passivem Verbindungsaufbau nicht verwendet.

datenlaenge Byteanzahl der maximal übertragenen Daten

timeout Überwachungszeit der Verbindung in Sekunden. Werden *n* Sekunden keine Daten ausgetauscht, wird die Sendeverbindung "gekappt". Die Empfangsverbindung wird analog dazu durch den Partner "gekappt".

 © IPKS GmbH 2007

6.9 Rechnerkopplung DUST 3966

Die DUST3966 Geräteemulation bietet grundsätzlich zwei Betriebsarten:

- Emulation der DUST 3966 über DF32 bzw. DF42
- Umlenkung der DUST 3966 Kommunikation auf TCP/IP

6.9.1 Emulation DUST 3966 über DF32/42

Die DUST 3966 Geräteemulation bearbeitet die Übertragungsprotokolle HDLC U/P, HDLC U/S und HDLC B.
Damit sind beispielsweise folgende Kopplungen realisierbar:

- M2000 - M2000 über HDLC B oder HDLC U und SINEC
- M2000 - IBM über SDLC und SINEC SNSNA
- M2000 - X.25 Netz über HDLC B und SINEC SNPV.

Nicht implementiert sind die WIKOP-Funktionen (Remote-Laden...)

6.9.1.1 Voraussetzung

Die Betriebssoftware für die eingesetzte Hardware wird von M2000 implizit installiert.
Zusätzlich zu dem Geräteeintrag (*device* = ..) in der Konfigurationsdatei *rpar.sys* (siehe 5.1.4 Kopplungen) erwartet die DUST 3966 Geräteemulation zusätzliche Parameter in einer Parameterdatei, deren Namen und Verzeichnis frei wählbar sind und im Geräteeintrag zugeordnet werden.

6.9.1.2 Parametrierung

Die anwendungsspezifischen Parameter werden in einer ASCII-Datei hinterlegt. Die Parameterzeilen können in Gross- und Kleinbuchstaben notiert werden, Kommentare werden mit " # " eingeleitet und gelten bis zum Zeilenende.

Folgende Parameter sind für beide Protokollarten einstellbar, wobei alle Angaben zu den Parametern (einschließlich Default-Einstellungen) der Beschreibung zum jeweiligen COMSOFT-Treiber (HDLC-UNC oder HDLC-BAC) zu entnehmen sind:

PATH *pfadname*	Verzeichnis mit COMSOFT-Komponenten
BOARD *nr*	Board-Nummer
UNIT *nr*	Schnittstellen-Nummer auf dem Board
DU_MODUS *wert*	Bitweise Einstellung von Sonderfunktionen mittels eines 32-Bit-Wertes:
Bit 0 = 0	Die Baugruppe wird beim Start der Emulation einmalig mit Treibern versorgt und parametriert.
Bit 0 = 1	Die Baugruppe wird bei jedem ORG-Anstoss "Basisparametrierung" mit Treibern versorgt und parametriert.
Bit 1 = 0	keine 3277-Verbindung zu IBM
Bit 1 = 1	3277-Verbindung zu IBM
Bit 2 = 0	Eine HDLC-Verbindung wird ausschließlich mit dem COMSOFT-Treiber Aufruf „CLOSE_UNIT“ abgebaut.
Bit 2 = 1	Eine HDLC-Verbindung wird mit dem COMSOFT-Treiber-Aufruf „CLOSE_UNIT“ und anschließender Treiberparametrierung abgebaut.

Beim Einsatz einer DF42 Baugruppe ist die folgende Zeile zu
aktivieren
OSFILE OS42.BIN

Beim Betrieb einer DUST 3966B, DUST 3966UP oder einer DUST 3966US über eine COMSOFT-Baugruppe muss derzeit in Parameter DU_MODUS Bit 2 gesetzt werden, damit nach dem logischen Verbindungsaufbau auch die physikalische Verbindung abgebaut wird (Anpassung an aktuelles Verhalten der COMSOFT-Treiber).

6.9.1.3 Protokollparameter für HDLC-UNC

Beim Betrieb einer DUST 3966UP oder einer DUST 3966US über eine COMSOFT-Baugruppe können ab COMSOFT-Treiber DHDLC.DRV C01 auch Modemsignale bearbeitet werden. Dazu sind die neuen Parameter MODEM und MODEM_TIMEOUT einzustellen. Neben dem COMSOFT-Treiber DHDLC.DRV sind auch die COMSOFT-Komponenten DHDLC.DAT, OS32.BIN und DFXX.SYS neu in das Windows-System einzubringen. Dabei muss DFXX.SYS in das Verzeichnis *%systemroot%\system32\drivers* eingebracht werden, anschließend ist Windows neu zu starten.

Folgende Protokollparameter sind zum Treiber HDLC-UNC einstellbar:

MODEM	YES | NO	
MODEM_TIMEOUT		
SNRMREP	*value*	(0..255)
NRZ	YES | NO	
CRC	0 | 1	
MASTER	YES | NO	
CLOCK	*value*	(0..3)
MULTIPOINTYES | NO		
CREDIT	*value*	(0..7)
IFRAMESIZE	*value*	(128..1024)
STATIONS	*value*	(1..32)
HDLCADR	*adr*	
TIMER1	*value*	
TIMER2	*value*	
BAUDRATE	*value*	

 © IPKS GmbH 2007

6.9.1.4 Protokollparameter für HDLC-BAC

Folgende Protokollparameter sind zum Treiber HDLC-BAC einstellbar:

DCE	YES | NO	
MODEMDEF	*value*	(0..15)
MODEM_I_TO	*value*	(0..255)
MODEM_I_OK	*value*	(0..255)
AUTORESTART	YES | NO	
SENDSABM	YES | NO	
REP	*value*	(0..255)
NRZ	YES | NO	
CRC	0 | 1	
RNRBRK	*value*	(0..255)
CLOCK	*value*	(0 | 1 | 4)
CREDIT_HDLC	*value*	(0..7)
V24V11	0 | 1	
IFRAMESIZE	*value*	(64..1024)
HDLCADRCON	*adr*	
HDLCADROWN	*adr*	
TIMER1	*value*	
TIMER2	*value*	
BAUDRATE	*value*	

6.9.2 Umlenkung der DUST 3966 Kommunikation auf TCP/IP

In der Parameterdatei eines der beiden beteiligten Rechner wird der Host-Name des Partnerrechners eingetragen, wie er auch in der Datei ...*hosts* eingetragen ist; dieser Rechner verhält sich später beim Verbindungsaufbau "aktiv". Auf dem anderen beteiligten Rechner wird als Host-Name ein Leerstring eingetragen; dieser Rechner verhält sich später beim Verbindungsaufbau "passiv".
Die Übertragung der Daten erfolgt verbindungsorientiert über das Level 4 Transportprotokoll TCP.
Die Portadresse ist auf beiden Rechnern identisch, es muss eine Zahl von mindestens 10000 angegeben werden.

Spezifische Anlauffunktionen sind nicht notwendig.

6.9.3 Umlenkung der DUST3964R Kommunikation auf TCP/IP

In der Parameterdatei der beiden beteiligten Rechner wird der Host-Name oder die Host Adresse des Partnerrechners eingetragen, wie er auch in der Datei ...*hosts* eingetragen ist. Die Übertragung der Daten erfolgt verbindungslos über das Level 4 Transportprotokoll UDP. Auf Anwendungsebene sollte ein Quittungsverkehr implementiert sein (beispielsweise wie innerhalb von SINEC VS).
Diese Übertragung eignet sich auch dafür, um einfache Kopplungen zu Fremdsystemen wie UNIX zu realisieren.
Die Portadressen für jeden Rechner müssen eindeutig sein, es muss eine Zahl von mindestens 10000 angegeben werden.
Spezifische Anlauffunktionen sind nicht notwendig.

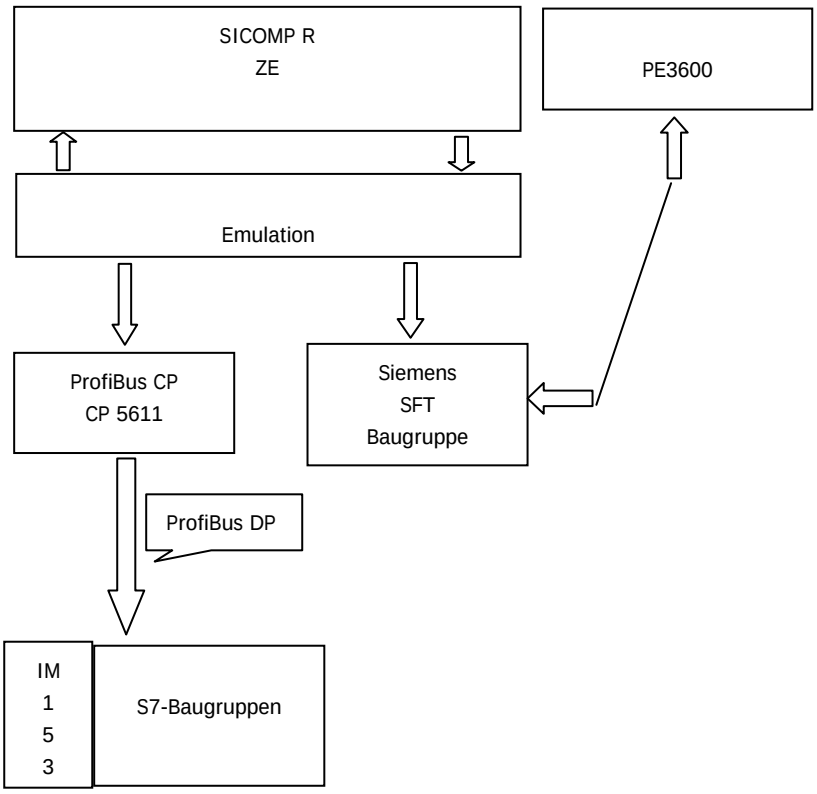
© IPKS GmbH 2007

6.10 Emulation PE3600 über ProfiBus oder SFT direkt

6.10.1 M2000

Eine SICOMP-R Konfiguration wird in einem M2000 System, mit dem Siemens ProfiBus als Schnittstelle zur Prozessperipherie, nachgebildet.
Entsprechend zu der bei SICOMP-R verwendeten PE3600 wird bei der Emulation die Verbindung zur Prozessperipherie durch S7-Baugruppen realisiert. Als Anschaltung wird eine ProfiBus-Baugruppe CP 5611 im M2000 Rechner eingesetzt. Diese Baugruppe stellt die Verbindung zwischen Windows/M2000 und der S7-Hardware her.
Eine zweite Variante ermöglicht die Verbindung zwischen Windows/M2000 und der PE3600 über die Siemens SFT Baugruppe direkt.

Damit ergibt sich die im folgenden Bild gezeigte Konfiguration.



6.10.2 Konfiguration *rpar.sys*

Die Parametrierung des Prozessbetriebes erfolgt zentral über die M2000 Konfigurationsdatei *rpar.sys*. In dieser Datei müssen die unten beschriebenen Einträge vorhanden sein.

Ein ‘device’-Eintrag zur Einbindung der PE3600-Aufrufe in die Emulation.

DEVICE = *anr,gnr,PE3600*

anr Anschlussstellennummer des Gerätes (Dezimalzahl)

gnr Gerätenummer (Dezimalzahl)

Ein Eintrag zur Zuordung der nachfolgenden Prozesssignalformer zu einer Hardwarebaugruppe. Dieser Eintrag ist gültig bis zum nächsten PSFDEV-Eintrag.

 © IPKS GmbH 2007

PSFDEV = *devtyp,devno,mode,polltime*

devtyp Baugruppentyp
PROFI = Profibusbaugruppe
SFT = SFT-Baugruppe
devno Baugruppennummer
PROFI 1.... dezimal
SFT 0...3 dezimal
mode Modus, hexadezimal, z. Zt. nur 0 sinnvoll

polltime Zykluszeit in Millisekunden, nach der alarmbildende (Eingabe-) Signalformer auf Veränderungen abgefragt werden.

Ein Eintrag für jede Grund- und Erweiterungssteuerung, sowie jeden Prozeßsignalformer mit einer Zuordung zu einer Adresse der vorher im PSFDEV-Eintrag definierten Baugruppe. Die Syntax ist sowohl abhängig von der zugeordneten Baugruppe, als auch vom Prozeßsignalformertyp. Nachfolgend die Syntax für die z.Zt. implementierten Steuerungen und Signalformer für den Profibus als zugeordnete Baugruppe.

Parametrierung einer Grundsteuerung GS3601-B:

PSF = GS3601-B,*sicAdr*,*AdressEbene,Spas,Reset,Poa*

Die Parameter in der Anweisung haben dabei folgende Bedeutung:

sicAdr Adresse der Steuerung unter SICOMP-R

AdressEbene Adressebene der Grundsteuerung, mögliche Einträge sind AE1 für Adressebene1 und AE2 für Adressebene2. Entspricht in ihrer Bedeutung den Schaltern S2 und S3.

Spas Adressierungsmodus der SP-Anschlussstellen. SPAS16 bedeutet, dass nur die Anschlussstellen 0 bis 15 adressierbar sind; SPAS22 bedeutet, dass auch die Anschlussstellen 16 bis 21 adressierbar sind. Der Eintrag entspricht in seiner Bedeutung dem Schalter S1.

Reset Dieser Eintrag bestimmt, ob die Steuerung per ZK rückgesetzt werden kann. Mögliche Einträge sind ZKRESET und NORESET. Entspricht in seiner Bedeutung dem Schalter S6.

Poa Dieser Eintrag bestimmt, ob die Steuerung eine POA auslösen kann. Mögliche Einträge sind POA und NOPOA. Entspricht in seiner Bedeutung dem Schalter S8.

© IPKS GmbH 2007

Nachfolgend werden die in der Grundsteuerung gesteckten Prozeßsignalformer parametriert:

Digitaleingabe DE3615-C

PSF = DE3615-C,*sicAdr*,SlaveNo,OffSet,Alarm

sicAdr Adresse des Prozeßsignalformers unter SICOMP-R

SlaveNo Nummer des Profibus-Slave.

Offset Offset-Adresse innerhalb des Profibus-Slave.

Alarm Dieser Eintrag bestimmt, ob die Steuerung alarmbildend ist und somit eine POA auslösen kann. Mögliche Einträge sind ALARM und NOALARM. Entspricht in seiner Bedeutung dem Schalter S8.

Digitalausgabe DA3621-B

PSF = DE3621-B,*sicAdr*,SlaveNo,OffSet

sicAdr Adresse des Prozeßsignalformers unter SICOMP-R

SlaveNo Nummer des Profibus-Slave.

Offset Offset-Adresse innerhalb des Profibus-Slave.

 © IPKS GmbH 2007

6.10.3 Beispiel

In der Parameterdatei *rpar.sys* sind die folgenden Anweisungen eingetragen.

```
device = 6,0,PE3600                                1.)

PSFDEV = PROFI,1,0,100                             2.)

PSF = 3601-B,0000,AE1,SPAS16,ZKRESET,POA           3.)

PSF = 3615-C,6000,3,0,ALARM                        4.)
```

1. Dieser Eintrag in der Parameterdatei definiert eine PE3600 als Gerät Nr. 0 an Anschlußstelle 6 .

2. Dieser Eintrag definiert eine Profibusbaugruppe (Nummer 1) mit Modus 0. Dies bewirkt, dass die Zugriffe für die nachfolgenden, durch PSF-Einträge definierten Prozeßsignalformer, über diese Baugruppe ausgeführt werden. Alle 100 Millisekunden werden alle alarmbildenden Eingabebaugruppen auf Veränderungen geprüft.

3. Dieser Eintrag definiert eine Grundsteuerung GS3601-B unter R-Adresse 0 in der Adressebene 1. Die SP-Adressierung ist nur von 0 bis 15 möglich. Rücksetzen durch die ZK sowie periphere Organisationsanforderungen sind möglich. Alle nachfolgend definierten Prozeßsignalformer bis zum nächsten Steuerungseintrag werden als in dieser Steuerung gesteckt angenommen.

4. Hier wird eine 32-Bit-Digitaleingabe unter R-Adresse 6000H definiert. Die Zugriffe werden auf den Profibus-Slave 3 und die Offset-Adressen 0 bis 3 abgebildet. Die Baugruppe ist alarmbildend, d. h. es werden bei Veränderungen POAs ausgelöst.

6.11 PE3600 – Siemens SFT Baugruppe

6.11.1 Beispiel

```
device = 6,0,PE3600                                1.)

PSFDEV = SFT,0,00000000                            2.)

PSF = 3601-B,0000,AE1,SPAS16,ZKRESET,POA           3.)
```

1. Dieser Eintrag in der Parameterdatei definiert eine PE3600 an Anschlußstelle 6, Gerätenummer 0.

2. Dieser Eintrag definiert eine SFT-Baugruppe (Nummer 0) mit Modus 0.

3. Dieser Eintrag definiert eine Grundsteuerung GS3601-B unter R-Adresse 0 in der Adressebene 1. Die SP-Adressierung ist nur von 0 bis 15 möglich. Rücksetzen durch die ZK sowie periphere Organisationsanforderungen sind möglich. Es ist nur eine PSF = ... Anweisung erforderlich.

Hinweis

In der Grundsteuerung 3601 muss zwingend die Baugruppe C71458-6008-A... eingesetzt werden, da sonst keine Verbindung zwischen SFT und PE3600 zustande kommt.

6.12 Schnittstellenkonzentrator

6.12.1 Allgemeines

In SICOMP Konfigurationen sind Drucker, Bildschirme und andere Peripheriegeräte normalerweise über serielle Leitungen an den Rechner angeschlossen. Bei größeren Anlagenkonfigurationen erfordert dies einen erheblichen Verkabelungsaufwand.
Mit M2000 lässt sich dieser Verkabelungsaufwand reduzieren, wenn mehrere serielle Peripheriegeräte zwar vom SICOMP Rechner entfernt aber dort dann örtlich zusammenstehen (z.B entfernte Betriebsgebäude, entfernte Maschinen, entfernte Arbeitsplätze usw.). Für die M2000 Schnittstellenemulationen DS075 (serieller Sichtgeräteanschluss), DR202 (serieller Druckeranschluss) und EAMX (PROMEA MX serieller Anschluss) gibt es Stellvertreterprogramme die auf entfernten, über LAN mit dem M2000 Rechner verbundenen, Windows Systemen ablaufen können. Diese seriellen Schnittstellen können damit vom M2000 System auf einen entfernten, über LAN verbundenen Windows PC 'ausgelagert' werden. Die Peripheriegeräte werden dann an die seriellen Schnittstellen des entfernten Windows PC angeschlossen.

Auf dem M2000-Rechner ist je ausgelagerter Schnittstelle die zugehörige Parameterzeile in der Datei *mpar.sys* zu ändern. Auf dem entfernten Rechner ist je Schnittstelle ein Stellvertreterprozess zu starten.

Alle notwendigen Schnittstellen-Parametrierungen werden dem entfernten Rechner nach erfolgtem Verbindungsaufbau und bei jeder Neuparametrierung - beispielsweise nach einem Neustart des SICOMP-Systems - übermittelt. Damit kann jeder der beteiligten Rechner einzeln gebootet werden, Startreihenfolgen o.ä. sind nicht zu beachten. Bei nicht aufgebauten Verbindungen verhalten sich die lokalen Geräteprozesse ähnlich wie bei ausgeschalteten Endgeräten.

Für jede ausgelagerte Schnittstelle werden **zwei** TCP/IP-Verbindungen zwischen dem M2000-Rechner und dem entfernten Rechner aufgebaut, dabei ist der M2000-Rechner der passive Partner.

6.12.2 *mpar.sys* Parametrierung

Für die Geräte DR202, DS075 und EAMX kann als Gerätename statt "\\.\COMxx“ auch "REM:y“ angegeben werden. y steht für die Portnummer des ersten TCP/IP-Sockets, den der jeweilige Prozess auf dem M2000-Rechner einrichtet. Es wird ein zweiter Socket mit der Portnummer y+1 eingerichtet.

Beispiel für Einträge in *mpar.sys* :

```
DEVICE = C0C8, DS075, "REM:22000";
DEVICE = C020, DR202, "REM:23000";
DEVICE = C040, EAMX,  "REM:24000", 1, 0, 0;
```

 © IPKS GmbH 2007

6.12.3 Stellvertreterprozesse auf dem entfernten Rechner

Es wird gerätetypabhängig einer der drei neuen Prozesse RC_DRUA.EXE, RC_DS075.EXE und RC_EAMX.EXE gestartet. Die Aufrufe haben alle die gleiche Form.

RC_DRUA.EXE \\.\COMxx SocketOwn Host:SocketCon
RC_DS075.EXE \\.\COMxx SocketOwn Host:SocketCon
RC_EAMX.EXE \\.\COMxx SocketOwn Host:SocketCon

SocketOWN ist die Portnummer des ersten eigenen Sockets, der Port SocketOwn+1 wird auch verwendet.

Host ist der Name des M2000-Rechners laut Datei: *%systemroot%system32\drivers\etc\hosts*.

SocketCon ist die Portnummer des ersten auf dem M2000-Rechner verwendeten Sockets.

COMxx ist die serielle Schnittstelle auf dem entfernten Rechner.

Auf dem entfernten Rechner sollten zusätzlich zu den drei Prozessen auch die Prozesse SIC_MELD.EXE und WAIT4.EXE installiert und in den Suchpfad aufgenommen werden.

Die Prozesse geben ihre Meldungen in ihrem Fenster und im Ereignisprotokoll aus. Die Prozesse beenden sich durch das Schließen des zugehörigen Fensters oder durch Abmelden des Benutzers bei WINDOWS.

Beispiel für einen Startbatch auf dem entfernten Rechner:

```
REM
REM Warten auf WINDOWS
REM
wait4

REM
REM Einstellen der Schnittstelle COM10 für DSSK20
REM
mode \\.\COM10 ...

REM
REM Starten des Stellvertreterprozesses für DSSK20
REM
start “DSSK20“ /min rc_ds075.exe \\.\COM10 22000 PCSIC1:22000

REM
REM Einstellen der Schnittstelle COM11 für DRUA21
REM
mode \\.\COM11 ...

REM
REM Starten des Stellvertreterprozesses für DRUA21
REM
start “DRUA21“ /min rc_drua.exe \\.\COM11 23000 PCSIC1:23000

REM
REM Einstellen der Schnittstelle COM12 für DRUA22
REM
mode \\.\COM12 ...

REM
REM Starten des Stellvertreterprozesses für DRUA22
REM
start “DRUA22“ /min rc_eamx.exe \\.\COM12 24000 PCSIC1:24000
```

6.13 OP31 Emulation

6.13.1 Inbetriebnahmehinweise zum Laufwerk SONY RMO-S580

6.13.1.1 Funktionsschalter an der Rückwand

A: Paritätsprüfung
= 0: SCSI-Paritätsprüfung
= 1: keine SCSI-Paritätsprüfung

B: reserviert

C: Steuerung des Schreib-Cache
= 0: Schreib-Cache aktivieren
= 1: Schreib-Cache deaktivieren

D: Manueller Disk-Auswurf
= 0: Auswurf mit der EJECT-Taste möglich
= 1: kein Auswurf mit der EJECT-Taste möglich

E: SCSI1/SCSI2
= 0: SCSI2
= 1: SCSI1

F: Abschluss
= 0: Interner Abschluss deaktiviert
= 1: Interner Abschluss aktiviert

G: Automatischer Motorstart
= 0: Beim Einlegen einer MO-DISK wird der Spindelmotor gestartet
= 1: Beim Einlegen einer MO-DISK wird der Spindelmotor nicht gestartet

H: Reserviert

6.13.1.2 Inbetriebnahme

- Netzkabel anschließen
- SCSI-Kabel anschließen
- SCSI-ID einstellen (Rückwand)
- Abschluss SCSI Kabel (Falls kein weiteres Gerät, Schalter F=1)
- Laufwerk einschalten
- Windows neu starten

© IPKS GmbH 2007

6.13.2 Wichtige Hinweise

Beim Start von Windows muss das MO-Laufwerk eingeschaltet sein.

Windows vergibt den nächsten freien Laufwerksbuchstaben für die MO. Wurde dieser Laufwerksbuchstabe schon für Netzwerklaufwerke benutzt, so kann keine Verbindung zu dem Netzwerklaufwerk unter diesem Buchstaben hergestellt werden. Die Verbindung ist aufzuheben und mit einem anderen Buchstaben wieder herzustellen.

Die Device-Nummern sind bei der ORG-Generierung, mit null beginnend, ohne Lücke anzugeben.

Es wird geprüft, ob das angegebene "PhysicalDrive" vom Typ *RemovableMedia* ist. Ist dies nicht der Fall, so wird keine Bearbeitung von M2000 durchgeführt.

Das MO-Laufwerk wird von M2000 als "PhysicalDrive" angesprochen werden. Dies ist nur mit Administrator-Rechten möglich.

Bei Erweiterungen von M2000-Installationen um die OP31 Option sind die Administrator-Rechte ggf. neu zu vergeben.

Wichtig! Bitte beachten Sie, dass das OP31 im M -ORG als P20PSD generiert ist. Nur dann wird mit DRIVE OFF auch der Datenträger ausgeworfen. Wenn das OP31 als Festplatte generiert ist, darf kein Datenträgerwechsel vorgenommen werden.

6.13.3 Datenträger

Es sind nur die Medien mit 600MB bei 512 Byte/Sektor zugelassen. Die Sony-Bezeichnung lautet:

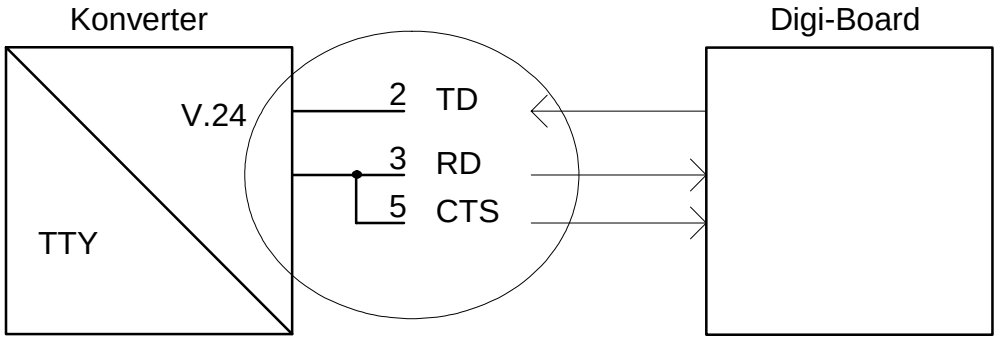
- EDM-600B
- EDM-1DA1s

© IPKS GmbH 2007

6.14 UNKLAR Erkennung bei seriellen Schnittstellen

6.14.1 TTY / V.24 Schnittstellen

Beim Betrieb über einen TTY / V.24 Konverter ist eine Brücke V.24-seitig von Pin 3 nach Pin 5 (RD - CTS) erforderlich. Da die Standard-Kabel zwischen Digi-Board und Konverter in der Regel vergossen sind, muss die Brücke im Konverter eingelegt werden. Das Signal CTS darf hierbei nicht zusätzlich vom Konverter beschaltet sein.



In der Parameterdatei (*mpar.sys/rpar.sys*) wird für die Geräte MX, Drucker, DS075 und 3974R die UNKLAR Erkennung über den Parameter *flags* bzw. *upar* eingestellt.

Achtung! Die oben beschriebenen Massnahmen gewährleisten eine UNKLAR-Erkennung für den Fall, dass die Empfangsdatenleitung (aus Sicht des Digi-Board) nach log. 0 (+12 Volt) wechselt. Für den Fall einer Linienstrom-Unterbrechung wird diese Bedingung vom Konverter erfüllt, wird jedoch z. B. das Kabel zwischen Digi-Board und Konverter abgezogen (oder fällt der Konverter aus), ist eine sichere Erkennung nicht gewährleistet, da dann der RD-Eingangspegel undefiniert ist. Eine wirklich saubere Erkennung könnte nur durch Einbau der Brücke in einem Adapter direkt am Digi-Board (oder im Digi-Board) mit einem zusätzlichem Pull-Up-Widerstand (RD gegen +12 Volt) realisiert werden.

6.14.2 V.24-Anschluss

Wenn das Endgerät direkt an die V.24 Schnittstelle angeschlossen wird, muss es das Signal DSR (Pin 6) zur Verfügung stellen.
In der Parameterdatei (*mpar.sys/rpar.sys*) wird für die Geräte MX, Drucker, DS075 und 3974R die DSR Abfrage über den Parameter *flags* bzw. *upar* eingestellt.

Achtung! Die oben beschriebenen Massnahmen gewährleisten eine UNKLAR-Erkennung für den Fall, dass das Signal DSR (aus Sicht des Digi-Board) nicht logisch EIN (+12 Volt) ist (eine gezogene Steckleitung wird erkannt). Mit den Standard-Drucker-Kabeln ist diese Lösung jedoch nicht realisierbar (Ausnahme: DR 3918), da lediglich TD, RD und Ground verbunden sind und somit ein auswertbares Signal vom Drucker nicht zur Verfügung steht!

© IPKS GmbH 2007

6.15 Kompatibilitätseinschränkungen

6.15.1 zum M-Rechner

M2000 bietet einen weitestgehenden Ersatz eines SICOMP M Minicomputers. Bei der Konzeption und Implementierung wurde Wert darauf gelegt, eine nahezuvollständige Kompatibilität zum Originalsystem zu erreichen. Trotzdem gibt es einige Einschränkungen in der Kompatibilität, die aufgrund von Hardware-Restriktionen nicht mehr mit wirtschaftlichem Aufwand zu beseitigen sind oder aufgrund anderer Vorteile sogar erwünscht sind.
Nachfolgend werden die bekannten Einschränkungen aufgeführt.

Gleitpunktwerte

Bei der Übergabe von Gleitpunktwerten aus einem emulierten ORG in das Windows-System, in der Regel über PSD/PSAPI, werden die Mantissen unterschiedlich verarbeitet.
Original SICOMP K-Format: 56 Bit Mantisse
Microsoft DOUBLE-Format: 53 Bit Mantisse

Zeitverhalten

Der technische Fortschritt hat dazu geführt, dass eine Emulation auf der heutigen PC-Hardware wesentlich höhere Leistungen bringt, als die veraltete Originalhardware des SICOMP M Systems. Dies führt dazu, dass die SICOMP M Anwendungssysteme schneller ablaufen. Insbesondere können Plattenzugriffe aufgrund von Cachetreffern um den Faktor 50-100 beschleunigt werden.
Diese Inkompatibilität im Zeitverhalten ist ein erwünschter Nebeneffekt in einer Emulation. Bei einer sauber konzipierten Anwendungssoftware bereitet dies auch keine Probleme.

Zeitgeber

Durch die verwendete Standard-Hardware und den Einsatz von Windows sind bei zeitmodifizierten Aufrufen Einschränkungen hinsichtlich der Granularität der Uhrzeit und der kleinsten möglichen zyklischen Weckzeit gegeben. Die kleinste Zeiteinheit beträgt bei ZIG1, 10ms, bei ZIG2, 1ms (Auflösung).

VICOM

Die folgenden VICOM-Kommandos sind nicht realisiert:

ACTIVATE	ACTIVATE CHECKPOINT
APZ	ACTIVATE PROCESS BY ‘APZ’
CALCULATE	CALCULATE TWO OPERANDS
DEACTIVATE	DEACTIVATE CHECKPOINT (S)
DELETE	DELETE CHECKPOINT (S)
DFP	DEACTIVATE PROCESS BY ‘DFP’
GENERATE	GENERATE ECC OF MEMORY
INFORM	ESTABLISH CHECKPOINT (NO STOP)
INTERRUPT	ESTABLISH CHECKPOINT (INTERRUPT)
PASSWORD	ESTABLISH OR DELETE PASSWORD
PROTECT	PROTECTION BY PASSWORD
SHOW	DISPLAY CHECKPOINTS
STEP	CONTINUE PROGRAM FOR STEP (S)
*	END OF CHANGE-INC-MODE

Testfunktionen

TESTAS und SPAS werden nicht unterstützt. Das Kommando /STATI liefert keine aussagefähigen Werte, da aus Gründen der Performance nicht alle Statistik Register einer SICOMP ZE nachgebildet werden.

Floppy Disk

Die Datenstruktur auf einer, mit der Emulation betriebenen Diskette, entspricht einer vom Originalsystem erzeugten Diskette. Trotzdem ist ein Datenaustausch über dieses Medium nicht möglich, da sie heutigen Floppy-Controller die besondere Modulation der Spur 0 nicht mehr unterstützen.

DU06

Die Geräteemulation der DU06 unterstützt nicht die WIKO-Funktionen. (Remote-Laden)

© IPKS GmbH 2007

Weitere

Abweichungen sind auch bei folgenden Funktionen bekannt:

- Datenträgerwechsel (/DRIVEON und /DRIVEOFF)
- INITM (DELETE und DEL-PTR)

Datenträgerwechsel (/DRIVEON und /DRIVEOFF)

Unter Windows ist der Datenträgerwechsel nicht erkennbar.

Zum Schutz vor Datenverlust wird in M2000 folgendermaßen vorgegangen:

- Das Laufwerk wird bei eingelegter MO-Scheibe verriegelt. Ein Wechsel der MO-Scheibe ist nicht möglich.
- Diese Verriegelung wird nur nach /DRIVEOFF aufgehoben. Die MO-Scheibe kann gewechselt werden (Wartezeit bis 20 Sekunden).

Zum Wechsel der MO-Scheibe sind also immer die beiden Kommandos /DRIVEOFF und /DRIVEON durchzuführen.
(/DRIVEON wird von M-Benutzern manchmal weggelassen, ist jedoch nach ORG-M-Beschreibung der richtige Weg zum Wechseln von Datenträgern.)

:INIT:FORMAT...

Formatieren ist nur mit der Produktkennung 'P20PSD' und mit dem Zusatz FIRST zulässig:

```
:INITM:FORMAT PLSKxx-P20PSD FIRST
```

Alle anderen Formatiervarianten führen zum Formatieren der gesamten MO-Scheibe und damit zum Verlust aller Daten.

:INIT:INIT...

Beim Initialisieren sollte immer der Parameter LS angegeben werden, und zwar die gleiche Größe wie beim Strukturieren.

```
:INITM:INIT PLSKxx-P20PSD LS-xxxxx
```

Wird der LS-Parameter nicht angegeben, so meldet INITM unter Umständen 'Blockfactor > 1'.

:INITM:DELETE...

Das Löschen von logischen Teilgeräten wird von M2000 nicht durchgeführt. Die Funktion wird wirkungslos und ohne Anzeigen durchlaufen.

Es wird von M2000 auch kein DEL-PTR (Lösch-Zeiger) geführt.

Die weiteren INITM-Funktionen, wie z.B. Initialisieren des Datenträgers nach dem Löschen (nicht Parameter NOINIT), werden jedoch von INITM durchgeführt.

© IPKS GmbH 2007

6.15.2 zum R-Rechner

Die SICOMP R Emulation SIC-R bietet einen weitestgehenden Ersatz eines SICOMP R Minicomputers. Bei der Konzeption und Implementierung wurde Wert darauf gelegt, eine nahezu vollständige Kompatibilität zum Originalsystem zu erreichen. Trotzdem gibt es einige Einschränkungen in der Kompatibilität, die aufgrund von Hardware-Restriktionen nicht mehr mit wirtschaftlichem Aufwand zu beseitigen sind oder aufgrund anderer Vorteile sogar erwünscht sind.

Nachfolgend werden die bekannten Einschränkungen aufgeführt.

Gleitpunktwerte

Bei der Übergabe von Gleitpunktwerten aus einem emulierten ORG in das Windows-System, in der Regel über PSD/PSAPI, werden die Mantissen unterschiedlich verarbeitet.
Original SICOMP K-Format: 56 Bit Mantisse
Microsoft DOUBLE-Format: 53 Bit Mantisse

Zeitverhalten

Der technische Fortschritt hat dazu geführt, dass eine Emulation auf der heutigen PC-Hardware wesentlich höhere Leistungen bringt, als die veraltete Originalhardware des SICOMP R Systems. Dies führt dazu, dass die SICOMP R Anwendungssysteme schneller ablaufen. Insbesondere können Plattenzugriffe aufgrund von Cachetreffern um den Faktor 50-100 beschleunigt werden.
Diese Inkompatibilität im Zeitverhalten ist ein erwünschter Nebeneffekt in einer Emulation. Bei einer sauber konzipierten Anwendungssoftware bereitet dies auch keine Probleme.

Zeitgeber

Durch die verwendete Standard-Hardware und den Einsatz von Windows sind bei zeitmodifizierten Aufrufen Einschränkungen hinsichtlich der Granularität der Uhrzeit und der kleinsten möglichen zyklischen Weckzeit gegeben. Die kleinste Zeiteinheit beträgt 10ms (Auflösung).

Floppy Disk

Die Datenstruktur auf einer, mit der Emulation betriebenen, Diskette entspricht der vom Originalsystem erzeugten Diskette. Trotzdem ist ein Datenaustausch über dieses Medium nicht möglich, da die heutigen Floppy-Controller die besondere Modulation der Spur 0 nicht mehr unterstützen.

DUST3966

Die Geräteemulation der DUST3966 unterstützt nicht die WIKOP-Funktionen (Remote-Laden ...).

© IPKS GmbH 2007

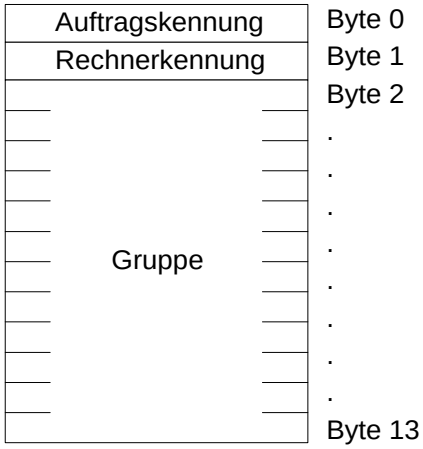
6.16 Kommunikationskanal SCU – SICOMP M/R Programme

DEVICE = *ioadr*,PSCU
DEVICE = *anr,gnr*,PSCU

ioadr IO-Adresse des Gerätes gemäß ORG-Generierung (4stellige Hexazahl).
anr,gnr Anschlussstellen-/GeräteNr. des Gerätes gemäß ORG-Generierung (Dezimalzahl)

M2000 richtet einen Kommunikationskanal zum SCU-Service ein und verbindet diesen mit dem im ORG generierten Sichtgeräte-Kanal DSSK. Das Anwenderprogramm kann über diesen Sichtgeräte-Kanal Telegramme mit dem SCU-Service austauschen (ORG-Aufrufe $STAUAL/$STEIAL).
Die Telegramme werden nicht explizit quittiert, es muss unter ORG-M/R immer ein Empfangsaufruf ($STEIAL) anstehen.

Aufbau der Anwendertelegramme:



Auftragskennung

SCU → Anwenderprogramm
- 0 Schaltvorgang erfolgt; die folgenden Bytes enthalten die aktuellen Informationen
- 1 Statusinformation; die folgenden Bytes enthalten die aktuellen Informationen

Anwenderprogramm → SCU
- 10 Schaltvorgang auslösen; Byte 1 ist leer, die Bytes 2 bis 13 enthalten den Namen der zu schaltenden Gruppe
- 11 aktuellen Status anfordern; die Bytes 1 bis 13 sind leer

Rechnerkennung — Kennung des prozessführenden Rechners (0...9) gemäß SCU-Parametrierung

Gruppe — Name der zuletzt geschalteten Gruppe gemäß SCU-Parametrierung

 © IPKS GmbH 2007

7 Zusatzprogramme

7.1 Programm PCCOPY

Mit dem Programm PCCOPY können Sie Bibliothekselemente oder ORG Dateien von einem SICOMP M/R System in das Windows-Dateisystem exportieren bzw. Windows-Dateien importieren:

- Beim Export werden SICOMP M/R Bibliothekselemente oder Dateien als Windows-Datei in einem Windows-Directory abgelegt.
- Beim Import wird eine Windows-Datei im SICOMP M/R Dateisystem als Datei oder Bibliothekselement abgelegt

PCCOPY ist ablauffähig im SICOMP M/R System.

Zur Installation des Programms PCCOPY unter AMBOSS/BS-M verfahren Sie wie folgt:

- Editieren Sie die Parameterdatei (*mpar.sys*), so dass der Pfadname einer der generierten Datenplatten auf die Datei *plsk.psd* zeigt.

  (im Lieferumfang enthalten, steht nach Installation des Anwendersystems zur Verfügung).

Beispiel:

```
device = 8400-0, FP0XX,1995, "..\Platten\plsk.psd";
device = 8,0, 3949, "..\Platten\plsk.psd";
```

- Starten Sie den Emulator.
- Das Programm PCCOPY ist auf der Datenplatte in der Bibliothek GSB und kann als PRP oder HRP geladen werden.
- Nachdem das Programm im System geladen ist, kann die Parameterdatei wieder auf den vorherigen Zustand zurückgeändert werden.
- Die Bedienung von PCCOPY wird über Fragezeichen erklärt:

```
:PCCOPY:?
COPY  COPY FILES AND LIBRARIES TO AND FROM DOS PARTITION
COPY device-lib.[element] TO dospath   Copy Library Element(s) to DOS-File(s)
COPY device-filename TO dospath        Copy File(s) to DOS-File(s)
COPY dospath TO device-lib.[element]   Copy DOS-File(s) to Library
COPY dospath TO device-filename        Copy DOS-File(s) to File(s)

Examples:
COPY PLSK1-QSB. TO C:\MYDIR            Copy all Library Elements to Directory MYDIR
COPY PLSK1-QSB.A* TO C:\MYDIR          Copy all Library Elements A* to Dir. MYDIR
```

Anmerkungen

Das Programm PCCOPY benutzt zum Lesen / Schreiben von Daten im ORG BIBEAS Aufrufe. PCCOPY macht selbst keine Umwandlung von Zeichen (Ausnahme siehe d). Damit kann es beim Kopieren von Daten (Windows -> ORG) zu folgenden Problemen kommen:

a) Das Quellelement enthält Umlaute wie ä (E4), Ä (C4), ü (FC) usw. entsprechend der Codierung einer unter Windows verwendeten Zeichentabelle. BIBEAS schreibt diese Zeichen entsprechend dem Quellelement in die Zieldatei. Diese Zeichencodierungen sind dann mit MEDIS nicht darstellbar. Entsprechend den ORG Konventionen müssten die Umlaute als ae, ue, geschrieben werden.

b) Das Quellelement enthält Horizontaltabulatoren. Die Tabulatoren werden beim Kopieren korrekt mit 09 im Zielelement abgelegt. Wenn das Zielelement im ORG über BIBEAS mit FILE / MEDIS im Modus 'nicht komprimiert' gelesen wird, so werden die Horizontaltabulatoren entsprechend den BIBEAS Regeln

durch eine entsprechende Anzahl Blank ersetzt und eine MEDIS Zeile kann größer als 80 Zeichen werden. BIBEAS kann HT nur dann richtig interpretieren, wenn sie auch nach seinen Regeln abgelegt wurden. Beim Einlesen im Modus - *komprimiert* - treten diese Probleme nicht auf. Um generell Probleme zu vermeiden sollten unter Windows im Quellelement die HT durch Blank ersetzt werden.

Beim Kopieren von Daten ORG -> Windows versucht PCCOPY mit dem Elementnamen eine Datei nach DOS Konventionen 8+3 einzurichten.Nach DOS Konventionen darf der Dateiname keine Blank enthalten.

c) Beim Kopieren von Daten ORG -> Windows wird ETX nach Linefeed (0A) kopiert. Die kopierten Dateien können dann mit einem Windows Editor angezeigt und bearbeitet werden. Beim zurückkopieren von Windows -> ORG wird Linefeed (0A) wieder nach ETX konvertiert.

d) Ist beim Kopieren von Daten ORG -> Windows das letzte Zeichen des Zielnamens ein $, dann wird unter diesem Namen abgespeichert.
Fehlt das $ Zeichen, so wird der Quell(Datei/BibElement)Name als Zielname benutzt.

Beispiel

```
:PCCOPY:COPY PLSK-BIB.ELNAME TO C:\ZDIR
```

Es wird unter ELNAME in ZDIR abgespeichert.

```
:PCCOPY:COPY PLSK-BIB.ELNAME TO C:\ZDIR\ZNAME.TXT$
```

Es wird unter ZNAME.TXT in ZDIR abgespeichert.

7.2 Programm MCSAVE.EXE

Das Programm MCSAVE ist ein Windows Programm, mit dem man MK82 Bänder in Windows importieren kann. Die Bedienmöglichkeiten von *mcsave.exe* werden angezeigt, wenn man das Kommando mcsave -? eingibt (siehe unten).

```
usage:  mcsave [-T geraet] [-van][-pxx][-r][-d]
-T      Geräteangabe in WINDOWS-Notation
-v      VolumeName wird als Dateiname verwendet, sonst plskxx
-pxx    erster Dateiname plskxx, xx=00...09, (sonst plsk00)
-a      alle Volumes des Magnetbandes einlesen
-n      Band nicht rückspulen
-r      Band nur rückspulen
-d      nur ein (1.) Datenträgeretikett lesen
-D      alle Datenträgeretiketten lesen
```

Anmerkung Bei Angabe der Parameter -d oder -D werden das erste oder alle Datenträgeretiketten gelesen, aber keine Daten kopiert.

Hinweis Die Funktionen von *mcsave.exe* werden von M2000 im Register:
`Allgemein – Daten importieren`
zur Verfügung gestellt.

 © IPKS GmbH 2007

7.3 Programm WAIT4.EXE

Das Programm WAIT4.EXE kann generell auf den Zugang zu Windows-Objekten warten und auf einen Zustand von M2000.

Es kann beispielsweise im Start Batch von M2000 aufgerufen werden und wartet dann bis alle Treiber und Dienste von Windows gestartet sind, bevor M2000 gestartet wird.

Es kann aber auch in Anwender Start-Batches aufgerufen werden, und wartet dann bis beispielsweise bestimmte Pipes von M2000 eingerichtet wurden.

Aufrufvarianten und Beispiele:

wait4 com99	Der Name com99 wird erweitert auf \\.\com99. Das Programm wartet bis auf \\.\com99 zugegriffen werden kann
wait4 \\.*pipe**coil2in*	Der Name \\.\.... wird unverändert übernommen. Das Programm wartet bis auf die Pipe *coil2in* zugegriffen werden kann.
wait4 file.1 -n	Durch den Parameter -n wird erzwungen, dass der Name *file.1* unverändert übernommen wird (siehe auch *wait4 com99*). Das Programm wartet bis auf die Datei *file.1* zugegriffen werden kann.

Alle Parameterangaben werden angezeigt, wenn *wait4 -?* oder *wait4 /?* eingegeben wird.

```
WAIT4 [name] [-tzz][-xzz][-n][-w][-r][-d][-?]
    name: name of the NT-devices to wait for Default: NT-Service-Database
    -tzz: wait Timeout (zz sec),      default = 60
    -xzz:delay zz sec before exit,    default = 2
    -n:   No name conversion          (*)
    -r:   test with Read access       (*)
    -w:   test with Write access      (*)
    -d:   Debug-Mode
    -?:   this screen                 (*) d.c by NT-Service-Database:
```

7.4 Programm SOLDIS

Das Windows Programm SOLDIS kann Daten (auch auf SICOMP M/R geschriebene) von MO Disk lesen und in Windows Dateien schreiben sowie Windows Dateien auf MO Disk schreiben.

Aufruf:
soldis [options] file1 file2 ...

options:
/s
Daten von file1 usw. auf die MO Disk schreiben (Voreinstellung)
/l
Daten von MO Disk in die Dateien file1 usw. schreiben
/d#
Angabe der Nummer # des physical Drives (1..n). Diese Angabe entfällt normalerweise, da SOLDIS die Laufwerksnummer automatisch ermittelt. Die Angabe wird aber notwendig, wenn z. B. mehrere MO Laufwerke am PC angeschlossen sind.
/type
Angabe des emulierten Plattentyps:
- /3941
- /3942 (Voreinstellung)
- /3945
- /3948a
- /3949

/n
keine Startmeldung ausgeben
/v
verbose: Laufzeitmeldungen zur Fehlersuche ausgeben

 © IPKS GmbH 2007

7.5 SicView, M2000 Festplatten Explorer

Festplatten von Original SICOMP Systemen werden unter dem Emulator als WINDOWS Dateien geführt. Jede dieser Dateien ist ein Binärabbild einer Originalfestplatte. Zugriff auf die innerhalb eines solchen Abbild liegenden Daten, ist normalerweise nur über das emulierte SICOMP System möglich.
Benutzer mit geringen ORG-M/R Kenntnissen haben erfahrungsgemäß Schwierigkeiten beim Umgang mit emulierten SICOMP Systemen. **SicView** ermöglicht es diesem Benutzerkreis, SICOMP Daten auf die unter Windows übliche Art zu betrachten.

Mit **SicView** lassen sich die Buchführung und der Inhalt eines SICOMP Datenträgers visualisieren (auch über Netz). Dadurch können Benutzer ohne SICOMP Kenntnisse auf SICOMP Daten zugreifen. **SicView** liest das binäre Abbild einer SICOMP Platte und stellt deren Struktur ähnlich wie der Windows-Explorer dar.

Leistungsmerkmale

- Darstellung des Inhalts einer SICOMP Platte.
- Anzeige von ORG-Dateien und Bibliothekselementen in einer HEX, ASCII und WORT Darstellung.
- Ausdruck und Druckvorschau von ORG-Dateien und Bibliothekselementen.
- Export von ORG-Dateien und Bibliothekselementen in normale Windows Dateien. Import von veränderten QS-Elementen/Dateien ist möglich, von GS-Elementen/Dateien nicht. (Export/Import nur mit PCCOPY)

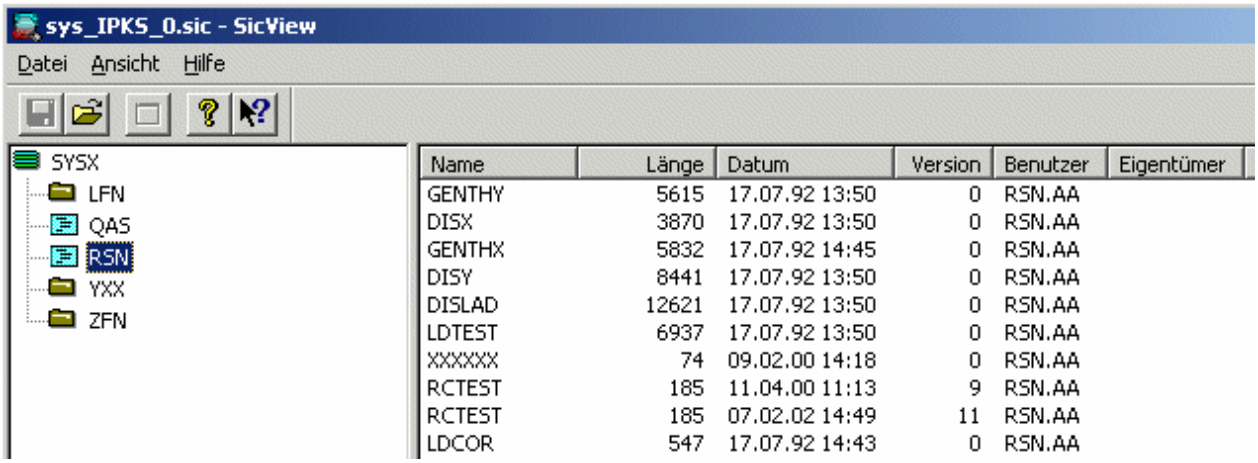
SicView greift nur lesend und offline auf eine (virtuelle) SICOMP Platte zu, so ist ausgeschlossen, dass Dateien innerhalb der Platte verändert werden.

Der Parameter *mode=share* muss in der Parameterdatei *mpar.sys/rpar.sys* eingetragen werden, um Probleme beim Zugriff auf einen exclusiv belegten Datenträger zu verhindern.

Zusätzlich ist die in **SicView** integrierte Funktionalität automatisierbar. Im Lieferumfang ist ein Beispiel enthalten, das über *Visual Basic for Applications* (VBA), den Zugriff auf SICOMP Daten aus einer Excel Tabelle demonstriert.
SicView ist im Lieferumfang von M2000 enthalten und muss im Dongle freigeschaltet sein. **SicView** kann dann über das M2000 Register :

`Allgemein` – Button

aktiviert werden.


sys_IPKS_0.sic - SicView
Datei Ansicht Hilfe
SYSX
LFN
QAS
RSN
YXX
ZFN
Name Länge Datum Version Benutzer Eigentümer
GENTHY 5615 17.07.92 13:50 0 RSN.AA
DISX 3870 17.07.92 13:50 0 RSN.AA
GENTHX 5832 17.07.92 14:45 0 RSN.AA
DISY 8441 17.07.92 13:50 0 RSN.AA
DISLAD 12621 17.07.92 13:50 0 RSN.AA
LDTEST 6937 17.07.92 13:50 0 RSN.AA
XXXXXX 74 09.02.00 14:18 0 RSN.AA
RCTEST 185 11.04.00 11:13 9 RSN.AA
RCTEST 185 07.02.02 14:49 11 RSN.AA
LDCOR 547 17.07.92 14:43 0 RSN.AA

8 Emulation PE F7

8.1 Hardwarestruktur

8.1.1 SICOMP M

Das SICOMP M System kann digitale und analoge E/A - Baugruppen in maximal 4 Baugruppenträgern EA 04 betreiben.
Jeder Baugruppenträger hat 19 freie Steckplätze. Wird einer dieser Steckplätze mit einer Anschaltung EG-AS 300-S bestückt, können darüber hinaus noch bis zu 4 Erweiterungsgeräte EG 180 bzw. EG 182 angesprochen werden, die jeweils 8 bzw. 6 Erweiterungsplätze bieten.
Als Maximalausbau können damit 4 * (18 + (3 * 8)) = 168 Baugruppen gesteckt werden. Geht man von der maximalen Ausstattung von 32 Ein- oder Ausgängen je Baugruppe aus, ergibt sich eine mögliche Zahl von 5376 Ein- / Ausgängen bzw. 672 EA - Bytes.

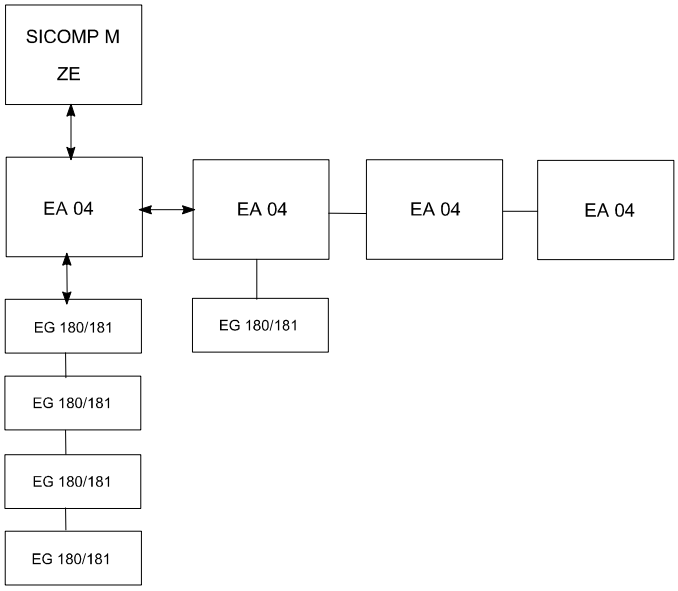


Bild 6: Prozessperipherie

 © IPKS GmbH 2007

8.1.2 M2000

Die im vorhergehenden Bild gezeigte SICOMP M Konfiguration wird in einem M2000 System, mit dem Siemens ProfiBus als Schnittstelle zur Prozessperipherie, nachgebildet.
Entsprechend zu den bei SICOMP-M verwendeten EA04-Baugruppenträgern wird bei der Emulation die Verbindung zur Prozessperipherie durch S7-Baugruppen realisiert. Als Anschaltung wird eine ProfiBus-Baugruppe CP 5611 in dem M2000-Rechner eingesetzt. Diese Baugruppe stellt die Verbindung zwischen Windows/M2000 und der S7-Hardware her.

Damit ergibt sich die im folgenden Bild gezeigte Konfiguration.

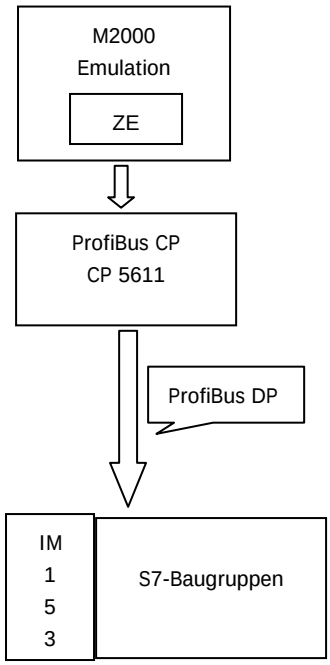


Bild 7: M2000 Prozessperipherie

8.2 Softwarestruktur

Das folgende Bild zeigt in einer Übersicht, wie SICOMP M Anwenderprogramme, die unter M2000 ablaufen, auf die Prozessperipherie zugreifen (Profibus).

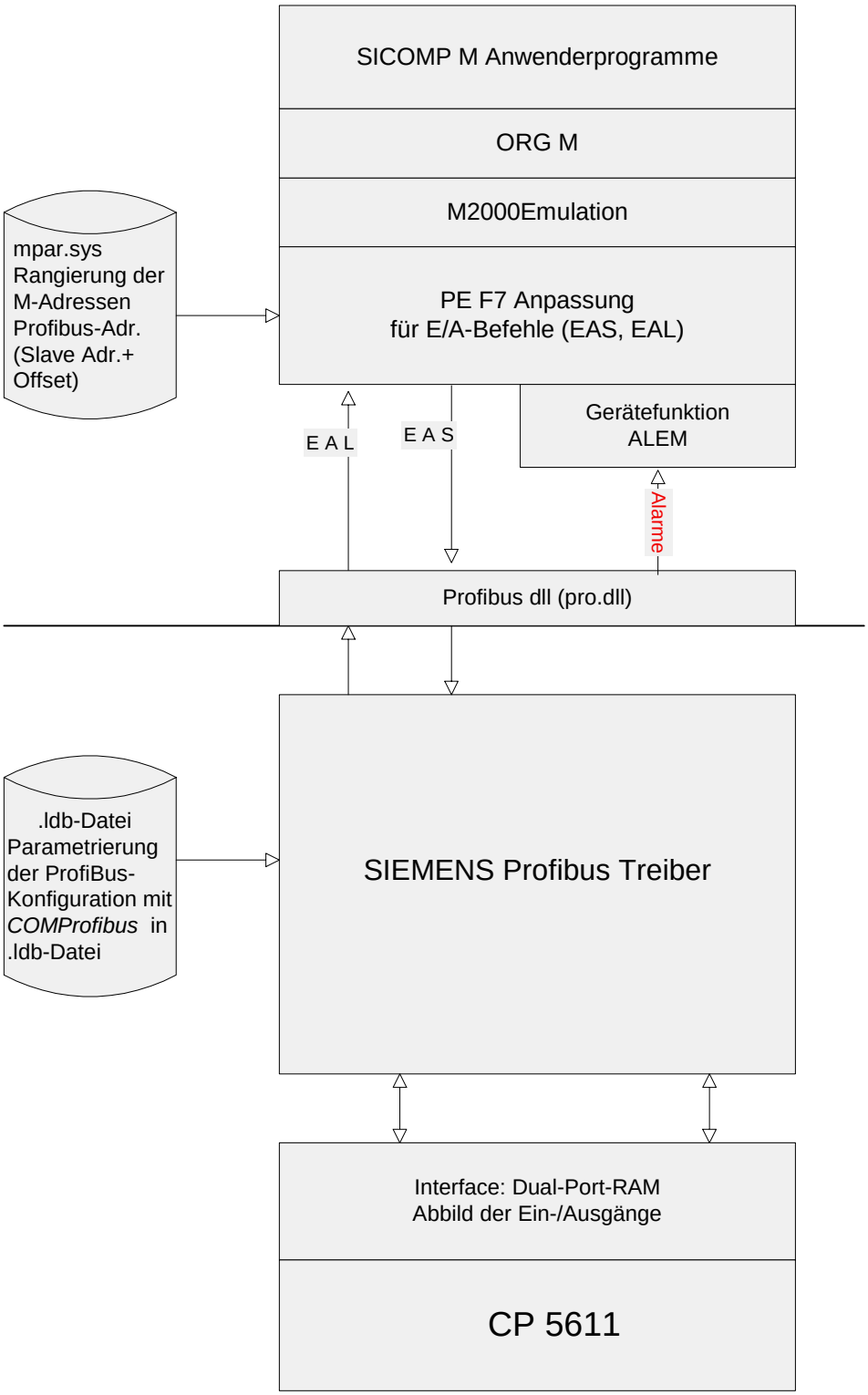


Bild 9:Zugriff über Profibus

© IPKS GmbH 2007

8.2.1 Zugriff auf die Prozessperipherie

M2000 enthält eine **PE F7** Geräteemulation für die E/A Zugriffe auf die Prozessperipherie. Diese Gerätemulation bearbeitet direkt die Prozessaufrufe, die nicht auf ein im ORG-M generiertes Prozessgerät führen. Prozessaufrufe, die das Prozessgerät ALEM verwenden, werden von einer M2000 Gerätefunktion **ALEM** bearbeitet. Ausgabebefehle an die Prozessperipherie werden vom ORG-M als EAS-Befehl an die PE F7-Peripherie weitergegeben. In der M2000 Geräteemulation werden diese Befehle an die ProfiBus-DLL (*pro.dll*) übergeben, die sie wiederum an den Siemens ProfiBus-Treiber weiterleitet. Auf diesem Weg gelangen die Ausgabedaten in den Dual-Port-RAM des CP 5611. Dieser gleicht zyklisch den Zustand des DPR mit dem der Hardware ab. Auf dem selben Weg werden in entgegengesetzter Richtung Eingaben aus dem DPR des CP den EAL-Befehlen des ORG zur Verfügung gestellt.

8.2.2 Alarmbearbeitung

Da der ProfiBus keine Alarmierung von Anwenderprogrammen bei Datenänderung bereitstellt, übernimmt die M2000 ProfiBus-DLL diese Funktion, indem sie zyklisch die relevanten Eingabedaten ausliest und bei Änderungen die Alarmierung des ORG veranlasst.

© IPKS GmbH 2007

8.3 Funktionsbeschreibung

8.3.1 Geräteemulation

Die PE F7 Geräteemulation bearbeitet in M2000 alle Aufrufe zur Prozessperipherie. Die Aufrufe zur Prozessperipherie sind unterteilt in Elementaraufrufe, die in einem direkten (Anwender-) Programmdurchlauf bearbeitet werden und in Aufrufe, gegebenenfalls mit einer Wartestelle, die auf ein im ORG-M generiertes Prozessgerät zugreifen.
Die Elementaraufrufe (EAL/EAS) werden in der PE F7 Geräteemulation selbst bearbeitet. Die Aufrufe auf ein Prozessgerät werden von der PE F7 Geräteemulation zur Bearbeitung an eine Gerätefunktion (Thread) weitergeleitet.

8.3.1.1 Elementaraufrufe

Die ORG-M-Elementaraufrufe wie BYTEINR, BYTAUSR usw. sind als Unterprogramme in der PE F7 Anpassung realisiert.

Die M2000 ZE-Emulation ruft nach Erkennen eines der genannten Befehle jeweils ein Unterprogramm in der *pro.dll* auf, das READ- und WRITE - Aufrufe direkt in die entsprechenden Siemens ProfiBus-Aufrufe umsetzt.

Anmerkung

Im SICOMP M System werden diese Aufrufe in der PE F7 Schnittstelle vorrangig bearbeitet und sind von anderen Ereignissen nicht unterbrechbar (Zustandwechselsperre ist gesetzt).

Mittels der M2000 Parametrierung über die Datei *mpar.sys* wird die im ORG-M verwendete PSF-Adresse in eine ProfiBus Adresse konvertiert. Das ProgrammzustandsRegister (PZR) wird entsprechend der Transferergebnisse des E/A-Befehls gesetzt.

8.3.1.2 Gerätefunktion

Die Bearbeitung der im ORG-M generierbaren Prozessgeräte übernimmt eine Gerätefunktion (Thread). Diese Gerätefunktion bearbeitet die ORG-M Aufrufe wie ALANM usw. zu dem Gerät ALEM.

Die M2000 ZE-Emulation stösst nach Erkennen eines EAL-Befehls für ein Gerät ALEM die Gerätefunktion an, die dann Aufrufe an die ProfiBus-DLL (*pro.dll*) absetzt und Aufrufergebnisse und Alarmereignisse an die ZE zurücksendet.

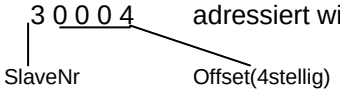
Mittels der M2000 Parametrierung über die Datei *mpar.sys* wird die im ORG-M verwendete PSF-Adresse in eine Treiberdatenadresse ProfiBus Adresse konvertiert. Der ProiBus-DLL werden mit dem jeweiligen Lese- oder Schreibaufruf die Daten ' ProfiBusadresse' und ' Werte (8 Bit)' übergeben. Der Returnwert wird entsprechend den ORG-M Vorgaben in die ORG-Aufrufanzeigen konvertiert.

8.3.1.3 Adressierung

Die Adresse besteht aus der jeweiligen Nr. des ProfiBus-Slave und einem Offset.

Beispiel

Adresse:$_{(hexa)}$
3 0 0 0 4 adressiert wird ByteNr. 4 im Adressraum des Slave mit der Nr. 3.



 © IPKS GmbH 2007

8.4 ProfiBus DLL

8.4.1 Allgemeines

Die Aufgaben der M2000 ProfiBus DLL sind folgende:

- im Anlauf der Emulation die Initialisierung der ProfiBus-Schnittstelle anstossen
- die Konvertierung der Lese-/Schreibaufrufe in die entsprechenden ProfiBus Aufrufe durchführen
- die Erzeugung von Alarmen nach Feststellung von Änderungen in den relevanten Eingangsdaten

8.4.2 Installation (ProfiBus)

Siehe:
Beschreibungen im Siemens SIMATIC-NET Paket

Anmerkung

Die ProfiBus Installation ist mit der aktuellen SIMATIC-NET CD durchzuführen und die verwendete Konfiguration zu parametrieren.
M2000 verwendet die CP_L2_1 – Schnittstelle zum Zugriff auf den ProfiBus.

8.5 Parametrierung

8.5.1 Konfigurationsdatei *mpar.sys*

Die Parametrierung des Prozessbetriebes erfolgt zentral über die M2000 Konfigurationsdatei *mpar.sys*. In dieser Datei müssen die nachfolgend beschriebenen Einträge vorhanden sein.
Ein ‘device’-Eintrag zur Einbindung der PE_F7-Aufrufe in die Emulation.

DEVICE = *ioadr*, ALEM, "PROFIBUS"

Ein Rangiereintrag für jedes Peripherie-Byte, aus dem Adresszuordnung, Typ und Einstellung des Signalformers hervorgehen.

Profibus

PSF=*sicAdr*,pro*adr*,*typ*,[*StaDyn*][*Flank*][,*Inter*][,*Analogm*]

Die Parameter in den Anweisungen haben dabei folgende Bedeutung.

sicadr Adress-Offset der Baugruppe relativ zu *ioadr*

proadr zugehörige Profibus-Adresse hexadezimal im Format xxxxyyyy, wobei xxxx die Slavenummer angibt und yyyy die Adresse der Baugruppe.

typ Typ der SICOMP-M Baugruppe in Anlehnung an die Syntax des SICOMP-M Testprogramms TESIG:

- DE8M 8 digitale Eingänge, belegt 1 Byte im SICOMP-M-Adressraum
- DE16M 16 digitale Eingänge, belegt 2 Bytes im SIC-M-Adressraum
- DA8M 8 digitale Ausgänge, belegt 1 Byte im SIC-M-Adressraum
- DA16M 16 digitale Ausgänge,belegt 2 Bytes im SIC-M-Adressraum
- AE8M 8 analoge Eingänge, belegt 16 Bytes im SIC-M-Adressraum, linksbündige Darstellung
- AEPB 8 analoge Eingänge, belegt 16 Bytes im SIC-M-Adressraum, rechtsbündige Darstellung
- AA8M 8 analoge Ausgänge, belegt 16 Bytes im SIC-M-Adressraum, linksbündige Darstellung

StaDyn Art der Baugruppe: ‘STATISCH’ oder ‘DYNAMISCH’, muss bei digitalen Eingabebaugruppen angegeben werden (DE8M oder DE16M).

Flank Bitmaske für die aktive Flanke bei dynamischen Baugruppen (nur bei StaDyn = DYNAMISCH)

- 0: zugehöriger Eingang reagiert auf negative Flanke
- 1: zugehöriger Eingang reagiert auf positive Flanke

Inter Der SICOMP-M Interrupt, der einer Baugruppe zugeordnet ist, muss bei digitalen Eingabebaugruppen angegeben werden (DE8M oder DE16M), falls diese alarmbildend sind.

Analogm Bei Angabe einer Analog-Baugruppe kann eine Umwandlung der Darstellungsmodi angegeben werden. Wird kein Analogmode angegeben, so werden die Werte in der Einerkomplementdarstellung übergeben.

- 2ER_KOMPLEMENT: Die Werte werden in der Zweier-Komplement-Darstellung übergeben (nur Analog-Eingabe).
- S7TRANS: Die Werte werden im S7-Format transparent weitergegeben. Es erfolgt keine Formatanpassung.
- 4_20MA: Formatanpassung zwischen der 4-20mA-Darstellung der S7 und der S5.

© IPKS GmbH 2007

8.5.2 Beispiel (ProfiBus)

In der Parameterdatei *mpar.sys* sind die folgenden Anweisungen eingetragen.

```
device = 2000,ALEM,“PROFIBUS“
psf = 0100,30010,AEPB,2ER_KOMPLEMENT          1.)
psf = 0040,40000,DA16M                        2.)
psf = 0020,40002,DE16M,STATISCH               3.)
psf = 0022,40002,DE8M,DYNAMISCH,F0            4.)
```

1. Unter der DeviceAdr 2100_{16} bis $210F_{16}$ werden 8 analoge Eingänge (AEPB) rechtsbündig adressiert und die Werte im 2er Komplement übergeben. Diese werden aus den EingangsByte 16-31 (3**0010**)$_{16}$ im ProfiBusSlave Nr. 3 (**3**0010)$_{16}$ gebildet.
2. Unter der DeviceAdr 2040_{16} werden 16 digitale Ausgänge angesteuert, die Werte werden im Slave Nr. 4, Byte 0 und 1 gebildet.
3. Unter der DeviceAdr 2020_{16} werden 16 digitale Eingänge angesteuert, die Werte werden im Slave Nr. 4, Byte 0 und 1 gebildet, Baugruppentyp = statisch.
4. Unter der DeviceAdr 2022_{16} werden 8 digitale Eingänge angesteuert, die Werte werden im Slave Nr. 4, Byte 2 gebildet, Baugruppentyp = dynamisch. Die Eingänge 0-3 reagieren auf negative, Eingänge 4-7 auf positive Flanke.

Anmerkung

Dynamische Eingänge werden durch die Emulation nachgebildet. In der S7 sind nur die standardmäßigen, statischen Eingabe-Baugruppen vorhanden.

9 Anhang

9.1 Allgemeine Hinweise

Floppy-Disk FLOP

Eine Diskette, die vom emulierten SICOMP System verwendet werden soll, muss auch mit SICOMP Systemmitteln formatiert werden.
Eine Diskette mit MS-DOS Format ist im emulierten System verwendbar, muss aber mit INIT eingerichtet werden. Danach ist sie unter DOS nicht mehr lesbar. Das Diskettengerät FLOP ist zum Datenaustausch über 5¼“ oder 3½“-Disketten zwischen verschiedenen emulierten SICOMP Systemen. Diese Disketten können auf einem originalen SICOMP System nicht gelesen werden.

FACO30

Hardcopy-Ausgaben können mit M2000 auf einen EPSON-kompatiblen Drucker am PC ausgegeben werden. Dazu muss in FACO30 auch EPSON eingestellt sein.

Drucker

Grundsätzlich sind beliebige Drucker an der parallelen oder seriellen Schnittstelle des PC anschließbar und betreibbar. Vorzugsweise sollte aber die serielle Schnittstelle verwendet werden, da der Windows Drucker für die parallele Schnittstelle mit Polling arbeitet und damit das Reaktionszeitverhalten von Windows negativ beeinflussen kann.
Es sollten nur Drucker verwendet werden, die standardmäßig im ASCII-Modus mit eigenem Zeichensatz betrieben werden können.

Druckersteuerzeichen

Druckersteuerzeichen (z.B. für Schriftartenumschaltung) werden von M2000 unverändert an den Drucker weitergegeben. Gegebenenfalls muss hier eine Anpassung von Anwendungsprogrammen erfolgen, wenn in Verbindung mit M2000 nicht der gleiche oder ein kompatibler Drucker zum Ursprungssystem verwendet werden kann.

Dongle

M2000 setzt die Existenz eines Dongle voraus, der an der parallelen Druckerschnittstelle gesteckt sein muss.

Erläuterungen zum Dongle-Etikett

M2000	Programmname
M2-xxx	Bezeichnung Basispaket
+ Optionen	Optionspakete
Dongle-Nr.:	
xxxxx	fünfstellige Donglenummer

DFCONS

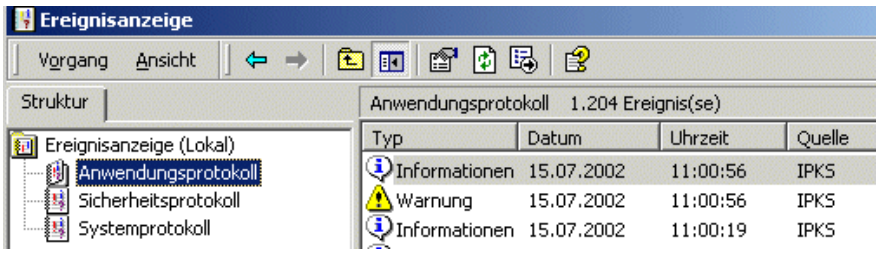
Im Gegensatz zum SICOMP System verfügt der PC nicht über einen gepufferten RAM-Speicher. Bei Verwendung des *IMAGE*-Parameters wird der Hauptspeicherinhalt beim Beenden der Emulation gesichert und beim nächsten Emulatorstart entsprechend vorbelegt. Damit ist die Rekonstruktion von DVS-Dateien bzw. DVS-Buchführungsdaten nach einer Spannungswiederkehr mittels DFCONS möglich.

 © IPKS GmbH 2007

9.2 Meldungen

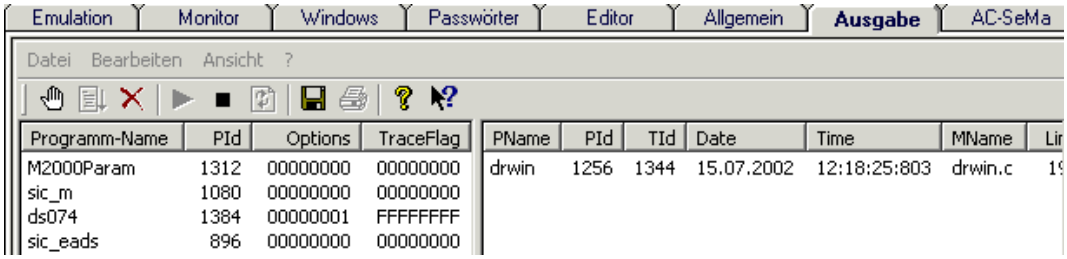
9.2.1 Start- / Ende- / Fehler - Meldungen.

M2000 schreibt seine Startmeldungen, Endemeldungen, Warnungen und Fehlermeldungen in das Ereignislogbuch von Windows (Anwendungsprotokoll). Der Aufbau der Meldungen entspricht den Windows Meldungen. Alle Meldungen von M2000 haben in der Spalte ' Quelle' die Kennung ' IPKS '.



9.2.2 Tracemeldungen.

9.2.2.1 TrcView-Fenster


Emulation Monitor Windows Passwörter Editor Allgemein Ausgabe AC-SeMa
Datei Bearbeiten Ansicht ?
Programm-Name PId Options TraceFlag
M2000Param 1312 00000000 00000000
sic_m 1080 00000000 00000000
ds074 1384 00000001 FFFFFFFF
sic_eads 896 00000000 00000000
PName PId TId Date Time MName
drwin 1256 1344 15.07.2002 12:18:25:803 drwin.c

Im linken Fenster von TrcView wird die Prozessliste ausgeben, im rechten die Traceliste. In der Prozess-Liste sieht man alle Prozesse, die die TRC-Schnittstelle(*trcdll.dll*) eingebunden haben.
Durch Doppelklicken auf einen Eintrag in der Prozess-Liste erhält man den Options-Dialog des angewählten Prozesses, über den man die Ausgabe der Trace-Meldungen steuern kann. Siehe auch: `TrcView Anwendungshilfe`
In der Trace-Liste werden die Trace-Meldungen angezeigt.

9.2.2.2 TrcView Leistungen

Das TRC-Modul stellt eine genormte, einfach zu handhabende und komfortable Trace- und Fehlermeldungsschnittstelle unter Windows zur Verfügung. Die Trace-Meldungen werden in einem Hauptspeicher-Umlaufpuffer erfasst, um Zeiteinflüsse durch die Trace-Ausgabe zu minimieren.

Die Traceschnittstelle bietet folgende Leistungen:

- Meldungs-Header mit Prozessname, Prozess-ID, Thread-ID, Datum/Zeit-Stempel, Modulname, Source-Zeilennummer
- Trace eines Puffers im Ascii- und hexadezimalen Format
- Wählen der Ausgabeart (Umlaufpuffer und/oder Datei) zur Laufzeit
- geringe Geschwindigkeitskeitseinbußen bei Meldungsausgabe in den Umlaufpuffer
- Einstellbare Größe der Meldungen und des Umlaufpuffers
- Speichern der Einstellungen in der Registratur (Button Sichern)
- Komfortables Anzeigeprogramm mit einer Liste aller Prozesse mit Trace-Ausgaben zum Ein- und Ausschalten der Meldungen und Anzeige des Umlaufpuffers

9.3 Muster der Parameterdatei *mpar.sys*

```
;-------------------------------------------------------------------------
;-------------------------------------------------------------------------
; M2000: M-Emulation: Beispielparameter, formatiert 80 Zeichen/Zeile
; Die Datei kann auch mit MEDIS editiert werden
;-------------------------------------------------------------------------
; keine Umlaute, MEDIS kompatibel !!!
;-------------------------------------------------------------------------
; Basisparameter
;-------------------------------------------------------------------------
; cpu = ze01                ; Typ der Zentraleinheit
; cpu = ze02
cpu = ze03
; zestart = 7               ; ZE Startverzoegerung = 7 Sekunden
; mode=popup                ; PopUp Fenster, weist auf interne Fehler hin
maxmem = 4032               ; Hauptspeicherausbau in kWorten
; image = "image"            ; HSP-Rettdatei
BOOT = 8200,8008
;       |   |_____________IO-Adresse der virtuellen Konsole
;       |_________________IO-Adresse der BOOT-Platte
;
;------- wenn Subadresse der BootPlatte, s. readme.wri !! --------
;
; Konsole lokal im Fenster
;-------------------------------------------------------------------------
device = 8008,DS074,"DSSK0"
;-------------------------------------------------------------------------
; DISIT-Bildschirm lokal im Fenster
;-------------------------------------------------------------------------
; device = 8008,3974MT,"DSSK0"
;-------------------------------------------------------------------------
; Konsole ohne Farb-Option an serieller Schnittstelle
; (mit setall.bat die COMx Schnittstellen parametrieren)
;-------------------------------------------------------------------------
;device = 8008,DS075,"\\.\COM2"
;  Eigener Dongle vorhanden
;device = 8008,DS075,"\\.\COM2",IPKS
;  Zentraler Dongle vorhanden
;device = 8008,DS075,"\\.\COM2",201,NONE,500
;  unklar Erkennung ein, bei V.24 Anschluss
;  Ueberparametrieren nicht moeglich
;  eigener Dongle vorhanden
;  Verzoegerungszeit 500 ms
;-------------------------------------------------------------------------
;Konsole mit Farb-Option an serieller Schnittstelle
;(mit setall.bat die COMx Schnittstellen parametrieren)
;-------------------------------------------------------------------------
;device = 8008,DS075F,"\\.\COM2"
;  Eigener Dongle vorhanden
;device = 8008,DS075F,"\\.\COM2",IPKS
;  Zentraler Dongle vorhanden
;-------------------------------------------------------------------------
; DISIT-Bildschirm an serieller Schnittstelle
; (mit setall.bat die COMx Schnittstellen parametrieren)
;-------------------------------------------------------------------------
;device = 8030,3974MTSER,"\\.\COM2"
;  Ueberparametrieren nicht moeglich
;  Eigener Dongle vorhanden
;device = 8030,3974MTSER,"\\.\COM2",1
;  Ueberparametrieren nicht moeglich
```

 © IPKS GmbH 2007

```
;  Eigener Dongle vorhanden
;device = 8030,3974MTSER,"\\.\COM2",1,IPKS
;  Ueberparametrieren nicht moeglich
;  Zentraler Dongle vorhanden
;device = 8030,3974MTSER,"\\.\COM2",0
;  Ueberparametrieren moeglich
;  Eigener Dongle vorhanden
;device = 8030,3974MTSER,"\\.\COM2",0,IPKS
;  Ueberparametrieren moeglich
;  Zentraler Dongle vorhanden
;-------------------------------------------------------------------------
; VDU2000 an serieller Schnittstelle
; (mit setall.bat die COMx Schnittstellen parametrieren)
;-------------------------------------------------------------------------
;device = 8038,VDU2000,"\\.\COM2"
;  Ueberparametrieren nicht moeglich
;  Eigener Dongle vorhanden
;device = 8038,VDU2000,"\\.\COM2",1
;  Ueberparametrieren nicht moeglich
;  Eigener Dongle vorhanden
;device = 8038,VDU2000,"\\.\COM2",1,IPKS
;  Ueberparametrieren nicht moeglich
;  Zentraler Dongle vorhanden
;device = 8038,VDU2000,"\\.\COM2",0
;  Ueberparametrieren moeglich
;  Eigener Dongle vorhanden
;device = 8038,VDU2000,"\\.\COM2",0,IPKS
;  Ueberparametrieren moeglich
;  Zentraler Dongle vorhanden
;-------------------------------------------------------------------------------
; DSSK ueber LAN, mehr als eine NetCard im PC
; (Netzwerksoftware (TCP/IP) muss in Windows installiert sein)
;-------------------------------------------------------------------------------
;DEVICE=8030,3974MTNETF,192.168.2.10:20010/0,IPKS
;DEVICE=8030,3974MTNETF,NetCard-LAN1:20010,IPKS
;-------------------------------------------------------------------------------
; Konsole ueber SOCKET
; (Netzwerksoftware (TCP/IP) muss in Windows installiert sein)
;-------------------------------------------------------------------------------
;device = 8008,DS074NET,22220
;  Eigener Dongle vorhanden
;device = 8008,DS074NET,22220,IPKS
;  Zentraler Dongle vorhanden
;-------------------------------------------------------------------------------
; DISIT-Bildschirm ueber SOCKET
; (Netzwerksoftware (TCP/IP) muss in Windows installiert sein)
;-------------------------------------------------------------------------------
;device = 8040,3974MTNET,22220
;  Eigener Dongle vorhanden
;device = 8040,3974MTNET,22220,IPKS
;  Zentraler Dongle vorhanden
;-------------------------------------------------------------------------------
; DS078-Bildschirm(TE2078 Emulation) ueber SOCKET
; (Netzwerksoftware (TCP/IP) muss in Windows installiert sein)
;-------------------------------------------------------------------------------
; device = 8410, DU78, "SOCKET", 20000,192.168.1.49:20001;
;-------------------------------------------------------------------------------
; Festplattenlaufwerke
; Achtung ! Die Eintraege fuer Festplatten muessen nach DevNr
; aufsteigend sortiert sein
;
```

```
; optional soll von anderen Programmen lesend auf die Platten
; zugegriffen werden (copy, SICVIEW, ...)
;MODE = SHARE
;-------------------------------------------------------------------------
; Festplattenlaufwerk PS048,PS049
;-------------------------------------------------------------------------
;device = 8200-0,PS048,"..\platten\plsk00"
;  64640 Sektoren / Ohne WRITE-THROUGH-Flag
;device = 8400-0,PS049,"..\platten\plsk10",1
;  25856 Sektoren / Mit WRITE-THROUGH-Flag
;-------------------------------------------------------------------------
; Festplattenlaufwerk FP0xx
;-------------------------------------------------------------------------
;device = 8100-0,FP0XX,64640,"..\platten\plsk00"
;  64640 Sektoren / Ohne WRITE-THROUGH-Flag
;device = 8200-0,FP0XX,25856,"..\platten\plsk10",1
;  25856 Sektoren / Mit WRITE-THROUGH-Flag
;device = 8400-0,FP0XXAE,64640,"..\platten\plsk00"
;  64640 Sektoren / Ohne WRITE-THROUGH-Flag
; mit ASCII-EBCDIC Konvertierung
;-------------------------------------------------------------------------
; Diskettenlaufwerk
;-------------------------------------------------------------------------
;device = 8104,FLOP,"\\.\A:"
;  Diskettenlaufwerk a:      15 Sektoren Format
;device = 8104,FLOP18,"\\.\B:"
;  Diskettenlaufwerk b:      18 Sektoren Format
;-------------------------------------------------------------------------
; Magnetbandkassettenlaufwerk MK82
; Der Treiber des Tandberg Laufweks muss in Windows
; installiert sein
; Der Zugriff auf das Laufwerk kann unter Windows mit der
; Funktion Verwaltung->Bandlaufwerk getestet werden.
;-------------------------------------------------------------------------
device = 8106,MK082,"\\.\TAPE0"
;  "Erstes" Laufwerk entsprechend der Numerierung in Windows
;device = 8106,MK082,"\\.\TAPE1"
;  "Zweites" Laufwerk entsprechend der Numerierung in Windows
;-------------------------------------------------------------------------
; Optische Platte OP31
; Fuer jedes an den PC angeschlossene Laufwerk nur einen
; Eintrag (es gibt keine Teilgeraete)
;-------------------------------------------------------------------------
;device = 8106,OP31,"\\.\PhysicalDrive0"
;  "Erstes" Laufwerk entsprechend der Numerierung in Windows
;device = 8106,OP31,"\\.\PhysicalDrive1"
;  "Zweites" Laufwerk entsprechend der Numerierung in Windows
;-------------------------------------------------------------------------
; Drucker
; (mit setall.bat die COMx Schnittstellen parametrieren)
;-------------------------------------------------------------------------
;device = 8020,DRUA,"LPT1"
;  Drucker am Parallelport
;  Schreibueberwachungszeit 20 Sek
;device = 8020,DRUA,"LPT1",20
;  Drucker am Parallelport
;  Schreibueberwachungszeit 20 Sek
;device = 8020,DRSPOOL,"LPT1"
;  Drucker am Parallelport
;  Betrieben ueber den WINDOWS-Treiber
;device = 8090,DRSPOT,"I:\DRFILE\DRDATEN",30
```

 © IPKS GmbH 2007

```
;  Druck in Datei, nicht DOSkompatibel
;  Schließen der Datei zeitgesteuert
;device = 8090,DRSPOTS,"I:\DRFILE\DRDATEN",30
;  Druck in Datei, DOSkompatibel
;  Schließen der Datei zeitgesteuert
;device = 8090,DRSPOTX,"\sicomp\spotx.par"
;  Druck in Datei, ueber Parameter gesteuert
;device = 8020,DRUA,"..\Drucker\drua0.txt"
;  Druckerausgaben in WINDOWS-Datei
;device = 8020,DRWIN,"drua0"
;  Druckerausgaben in WINDOWS-Fenster
;device = 8020,DR202,"\\.\COM7"
;  Drucker DR202 am seriellen Port
;  Schreibueberwachungszeit 20 Sek (default)
;  Keine Break-Erkennung
;device = 8020,DR202,"\\.\COM7",20,1
;  Drucker DR202 am seriellen Port
;  Schreibueberwachungszeit 20 Sek
;  Break-Erkennung
;device = 8020,DRCOM,"\\.\COM7",8,30,100
;  Drucker am seriellen Port
;  seriellen Port nicht parametrieren, mode Kommando gueltig
;  Schreibueberwachungszeit 30 Sek
;  Wartezeit 100 ms
;-------------------------------------------------------------------------
;  Datenuebertragungssteuerung DU03
;-------------------------------------------------------------------------
; device = A2C8,DU03,"NDIS",0;          fuer WindowsNT
; device = A2C8,DU03,"NDIS2000",0;    fuer Windows2000
;-------------------------------------------------------------------------
; Rechnerkopplung DU04
; (mit setall.bat die COMx Schnittstellen parametrieren)
;-------------------------------------------------------------------------
;device = A2C0,DU04,"\\.\COM2",11
;  Ueber serielle Schnittstelle
;  Prioritaet hoch / mit BCC
;device = A2C0,DU04,"\\.\COM2",00
;  Ueber serielle Schnittstelle
;  Prioritaet niedrig / ohne BCC
;device = A2C0,DU04,"\\.\DFXX0",00
;  Ueber COMSOFT DF32/DF42
;  Board 0 / Schnittstelle 0
;device = A2C0,DU04,"\\.\DFXX0",12
;  Ueber COMSOFT DF32/DF42
;  Board 1 / Schnittstelle 2
;-------------------------------------------------------------------------
; Rechnerkopplung DU05
;-------------------------------------------------------------------------
;device=8300,DU05,"\\.\DFXX0",512,g:\sicomp\du05.par;
; Ueber COMSOFT DF42
;device=8300,DU05,"SOCKET","WINNT1",22220,512,20000,10000,30000;
; GatewayFunction
; aktive Seite beim Verbindungsaufbau, Hostname eingetragen
;device=8300,DU05,"SOCKET","",22221,512,20000,10000,30000;
; passive Seite beim Verbindungsaufbau, Leerstring
;device=B400,DU05,"SOCKET","partner-11.12.13.1",20000,0;
; zwei Netzwerkkarten
;device=B400,DU05,"SOCKET","partner-LAN1",20000,0;
; Verbindung wird ueber Netzwerkkarte 1 aufgebaut
;device=B400,DU05,"SOCKET","partner-11.12.13.2",20000,0;
;device=B400,DU05,"SOCKET","-11.12.13.2",20000,0;
```

```
;device=B400,DU05,"SOCKET","partner-LAN2",20000,0;
; Verbindung wird ueber Netzwerkkarte 2 aufgebaut
;-------------------------------------------------------------------------
; Rechnerkopplung DU06
;-------------------------------------------------------------------------
;device=A2C0,DU06,"\\.\DFXX0",0,0,UP,1,512,0,0,0,du06up.par
;  Ueber COMSOFT DF32/DF42, Typ UP
;  Board 0 / Schnittstelle 0
;  Maximal eine Folgestation
;  IFRAME-Maximum 512 Byte
;device = A2C0,DU06,"\\.\DFXX0",0,0,US,0,512,0,0,0,du06us.par
;  Ueber COMSOFT DF32/DF42, Typ US
;  Board 0 / Schnittstelle 0
;  IFRAME-Maximum 512 Byte
;device =A2C0,DU06,"\\.\DFXX0",0,0,B,0,512,0,0,0,du06b.par
;  Ueber COMSOFT DF32/DF42, Typ B
;  Board 0 / Schnittstelle 0
;  IFRAME-Maximum 512 Byte
;device=A2C0,DU06,"SOCKET","winnt99",10000,B,1,512,30,5,0
;  Ueber SOCKET, Typ B
;  Partnerrechner WINNT99
;  Socket-Nummer 10000
;  IFRAME-Maximum 512 Byte
;device = A2C0,DU06,"SOCKET","",10000,B,1,512,30,5,0
;  Ueber SOCKET, Typ B
;  Partnerrechner ist aktiv
;  Socket-Nummer 10000
;  IFRAME-Maximum 512 Byte
;-------------------------------------------------------------------------
; Rechnerkopplung KS100
;-------------------------------------------------------------------------
;device = 8500,KS100,1,0,0,0
;  Betrieb ueber ersten CP1413
;device = 8500,KS100,2,0,0,0
;  Betrieb ueber zweiten CP1413
;-------------------------------------------------------------------------
; Rechnerkopplung UCP-2 (TCP/IP on board)
;-------------------------------------------------------------------------
;device = C100,UCP-2,0,0,0
;  Keine Einstellungen
;-------------------------------------------------------------------------
; Zeitgeber
;-------------------------------------------------------------------------
;device = 8010,ZIG1,"orgzeit"            ; Zeiteinstellungen relativ zu Windows
;device = 8050,ZIG1,"$NT_TIME$"          ; ORG+Windows Zeit synchron
;device = 8050,ZIG1,"$UTC_TIME$"         ; Zeit relativ zu GMT
;device = 8010,ZIG2;                     ; Zeit "millisekundengenau"
;-------------------------------------------------------------------------
; Prozesselement PE F7
;-------------------------------------------------------------------------
;device = 2000,ALEM,"\\.\PE_F7"
;  Keine Einstellungen
;  PSF =
;  Siehe Handbuch
;-------------------------------------------------------------------------
; Pseudogeraet fuer PSD
;-------------------------------------------------------------------------
;device = 8080,FTNT,"pipepsd",0
;  Ueber Pipe pipepsd
;  Nachrichtenlaenge maximal 4 kByte
;device = 8080,FTNT,"pipepsd",32
```

 © IPKS GmbH 2007

```
;  Ueber Pipe pipepsd
;  Nachrichtenlaenge maximal 32 kByte
;-------------------------------------------------------------------------
; Promea MX
; (mit setall.bat die COMx Schnittstellen parametrieren)
;-------------------------------------------------------------------------
;device = 8202,EAMX,"\\.\COM2"
;  Ueber serielle Schnittstelle
;  Keine Break-Erkennung
;-------------------------------------------------------------------------
; ES-Terminal
;-------------------------------------------------------------------------
;device = 8202,EAES,"\\.\COM2",34
;  Ueber serielle Schnittstelle
;  Eingabe transparent
;  Ausgabe transparent
;  9600 Baud
;device = 8202,EAES,"\\.\COM2",04
;  Ueber serielle Schnittstelle
;  Eingabe nicht transparent
;  Ausgabe nicht transparent
;  9600 Baud
;device = 8202,EAES,"\\.\COM2",03
;  Ueber serielle Schnittstelle
;  Eingabe nicht transparent
;  Ausgabe nicht transparent
;  4800 Baud
;-------------------------------------------------------------------------
; Watchdog
;-------------------------------------------------------------------------
;watchdog = "\\.\bello", 10, 15, 30, 30
;watch = STOP, Reset
;watch = STOP, RBORG
;watch = 8104, RBSIC
;watch = 8008, RBNT
;watch = 8008, RESET
```

9.4 Muster der Parameterdatei *rpar.sys*

```
;-----------------------------------------------------------------
;  M2000: R-Emulation: Beispielparameter, formatiert 80 Zeichen/Zeile
;     MEDIS kompatibel (keine Umlaute)
;
;
;  -----Basisparameter-----
;  Modell (R10/R10V/R20/R30)
   CPU=R10V
;
;  Hauptspeicherausbau in KW (64..1024)
   maxmem = 1024
;
;  Hauptspeicherpufferung
;  image = "image"
;
;  Hexaschalter (Boot-Adr.)
   BOOT=8025
;
;  Wartungsfeld (normalerweise nicht erforderlich)
;  WTF
;
;  Weitergabe von Befehlen an Geraeteprozesse ueber EA-U
;  EAU
;
;  -----Sichtgeraete-----
;  Das Sichtgeraet auf Anschlusstelle 1,0 ist virtuelle Konsole
;
;  device = 1,0,3974 ,"COM2"           ; serieller Anschluss der ZBE3974
;  device = 1,0,3974R,"COM2"        ; serieller Anschluss der ZBE3974R, DS074, DS075, TE2000 AX
;  device = 1,0,3974M,"\\.\COM2"      ; ZBE3974M, DS075 DISIT, TE2000 DX
;
;  device = 1,0,3974RT,"DSSE0"        ; lokale Emulation TE2000 AX, TE-KONS
;  device = 1,0,3974MT,"DSSE0"        ; lokale Emulation TE2000 DX
;  device = 2,0,3974MNET,"20001"      ; DISIT-Emulation ueber Netzwerk
;  device = 2,1,3974RNET,"20002"      ; TE2000 AX Emulation ueber Netzwerk
;
;  -----BelegungsModus fuer Platten-----
;  mode = share;                          ; = share, Belegmodus geteilt;
;                                         ; ohne mode =, Belegmodus exclusiv
;  -----Extern-Speicher-----
;  device = 5,0,3941 ,"plsk\disk0"
;  device = 5,0,3942 ,"plsk\disk0"
;  device = 5,0,3945A,"plsk\disk0"
;  device = 5,0,3945B,"plsk\disk0"
;  device = 5,0,3945C,"plsk\disk0"
;  device = 5,0,3946 ,"plsk\disk0"
;  device = 5,0,3948 ,"plsk\disk0"
;  device = 5,0,3948A,"plsk\disk0"
;  device = 5,0,3948B,"plsk\disk0"
;  device = 5,0,3949 ,"plsk\disk0"
;  device = 5,0,3949A,"plsk\disk0"
;  device = 5,0,3949B,"plsk\disk0"
;  device = 5,0,3949C,"plsk\disk0"
;  device = 5,2,3948 ,"plsk\plsk2"
;  device = 5,3,3948 ,"plsk\plsk3"
;
;  optische Platte OP11 am MEC-R Container
;  device = 5,0,OP11 ,"\\.\PhysicalDrive1"
;  MECR = 5
```

 © IPKS GmbH 2007

```
;
;  Magnetwechselplatte WP256 am MEC-R Container
;  device = 5,0,WP256 ,"\\.\PhysicalDrive2"
;  MECR = 5
;
;  Magnetbandkassette MK82 am MEC-R Container
;  device = 5,0,MK82 ,"\\.\Tape0"
;  MECR = 5
;
;  -----Zeitgeber-----
;  PROMEA-Zeitgeber
;  device = 1,3,ZIG1                    ;nur ORG Zeit, 1. Zeitaufruf wird verworfen
;  device = 1,3,ZIG1,"$NT_TIME$"        ;ORG+Windows Zeit synchron, 1. Zeitaufruf wird verworfen
;  device = 1,3,ZIG1,"$SYS_TIME$"       ;absolute Zeitwerte, 1. Zeitaufruf wird verworfen
;
;  device = 1,3,ZIG1X                   ;nur ORG Zeit, 1. Zeitaufruf wird abgesetzt
;  device = 1,3,ZIG1X,"$NT_TIME$"       ;ORG+Windows Zeit synchron, 1. Zeitaufruf wird abgesetzt
;  device = 1,3,ZIG1X,"$SYS_TIME$"      ;absolute Zeitwerte, 1. Zeitaufruf wird abgesetzt
;
;  Zeitgeber 3691A
;  device = 1,3,ALE1
;
;  -----Pseudogeraet fuer PSD / WINCC-PSD-----
;  device = 2,0,FTNT,"ftnt0",0,100,0
;  device = 4,0,FTNTX,"\\server1\pipe\ftnt0",7,200,1
;  device = 2,0,FTNTD,"ftntd0",0,512,0
;  device = 2,3,FTNTDX,"\\server1\pipe\ftntd0",8,512,0
;  device = 2,0,WINCC,"wincc0",8,1024,0
;
;  -----SonderPromea MX1-----
;  device = 3,0,EAMX1,"\\.\COM13",70,20,0x9
;
;  -----ES-Terminal-----
;  device = 1,1,EAES,"\\.\COM2",0x40
;
;  -----Floppy-----
;  device = 3,0,3944,"\\.\A:"
;
;  -----Drucker-----
;  Druckerbetrieb an der Parallel-Schnittstelle
;  bei Angabe von crlf wird bei alphanumerischer Ausgabe ein LF durch CR/LF
;  ersetzt (wichtig z. B. bei Laserdruckern)
; device = 1,1,DRUA,"LPT2",crlf
;  Druckausgabe in WinNT Datei
; device = 1,2,DRUA,"\sicomp\trc\drfile
;  serielle Drucker am seriellen Port
; device = 2,0,DRCOM,"\\.\COM11",3915,,100
;  serieller Drucker an Port 11 vom Typ 3915
;  Schreibueberwachungszeit 20 sek
;  Verzoegerungszeit 100 ms
;  Druckausgabe in ein Windows Fenster
; device = 4,0,DRWIN,"DRUA4"
;  Druckausgabe ueber Windows Druckertreiber
; device = 4,1,DRSPOOL,"LPT1"
;  Druck in Datei, ueber Parameter gesteuert
; device = 2,1,DRSPOTX,"\sicomp\spotx.par"
;
;  -----Kopplungen-----
;  DUST3964R ueber serielle Schnittstelle
; device = 7,0,3964R,"\\.\COM11",011
;  DUST3964R ueber TCP/IP
```

```
; device = 7,0,3964R,"SOCKET",22223,winnt7:22225
; device = 7,0,3964R,"SOCKET",22223,220.250.1.155:22225
;  DUST3965R ueber DF32/DF42
; device = 7,1,3965R,"\\.\DFXX0",0,0,512
;  DUST3965R ueber TCP/IP
; device = 7,1,3965R,"SOCKET","WINNT7",22220,512
;  DUST3966 ueber DF32/42
; device = 7,1,3966,"\\.\DFXX0",0,0,US,0,512,0,0,0,g:\sicomp\3966us.par
;  DUST3966 ueber TCP/IP
; device = 7,1,3966,"SOCKET","WINNT7",22223,US,1024,0,0,0
;  DUST 3961 ueber PARIF
; device=1,1,3961,"SOCKET",HOSTNT:22227,HOSTRR:22229,1024,0;
;  ----- Emulation KS N16 -----
;
; device = 7,0,CS275,"\\.\NATDEV",300
```

 © IPKS GmbH 2007

9.4.1 Muster des StartBatch *emu_strt.bat*

```
@echo off

REM M2000-Startbatch (Muster fuer Anwender)

REM Warten auf Ende des Windows-Hochlaufs
REM wait4                - wartet bis alle Treiber und Dienste von WINDOWS gestartet sind

REM Arbeitsverzeichnis einstellen
REM %m2000disk%          - die Variable ist nur gültig, wenn die Anwender-Systemumgebung installiert wurde
REM cd %m2000dir%\trc    - die Variable ist nur gültig, wenn die Anwender-Systemumgebung installiert wurde
k:                       - einstellen der Anwender-Partition
cd \m2000\sicomp\trc     - einstellen des Start-Verzeichnis

REM Einstellen der seriellen Schnittstellen
call  ..\setall.bat      - falls serielle Geräte betrieben werden, muss dieser Batch angepasst werden

REM M-Emulator starten mit DOS Fenster
REM start "IPKS_SIC-M" /min sic_m ..\mpar_IPKS.sys – startet die Emulation ( M ) mit einem DOS Fenster(minimiert)
                                                     - mit den Parametern der Konfig.Datei mpar_IPKS.sys

REM M-Emulator starten ohne DOS Fenster
sic_m ..\mpar_IPKS.sys   - startet die Emulation ohne DOS Fenster

REM R-Emulator starten mit DOS Fenster
REM start "EMULATOR" /min sic_r ..\rpar.sys   - wie oben, aber für ( R )

REM R-Emulator starten ohne DOS Fenster
REM sic_r ..\rpar.sys    - wie oben, aber für ( R )
```

9.5 M2000 Fehler-MeldeFormular

IPKS
Prozess-Software-Entwicklungs GmbH
H. Großheimann
Arnold-Dehnen-Straße 48
47138 Duisburg
Fon: 0203-410966-12
Fax: 0203-410966-25

eMail: support@ipks.de

Absender	
Firma	
Name	
Dienststelle	
Anschrift	
Telefon	
FAX	
eMail	
Datum	

Version M2000		Dongle-/Serien-Nr.	

SICOMP M Emulation		Windows	Version	
SICOMP R Emulation		Servicepack	Version	

TE2000 AX		Version	
TE2000 DX		Version	

M2000 Optionen	

Fehler bereits telefonisch gemeldet?

Ja [ ] am: [ ] an: [ ]

Beschreibung des Fehlers:

Folgende Unterlagen werden beigestellt:
siehe auch: Fehler-Analyse durch IPKS GmbH

© IPKS GmbH 2007

9.6 Fehler-Analyse durch IPKS GmbH

Um die Fehleranalyse bei Problemen mit der Emulation zu vereinfachen, müssen dem IPKS-Systemspezialisten vom Anwender folgende Unterlagen (Daten) als Dateien (.zip Archive) zur Verfügung gestellt werden.

1. Die Parameterdateien *mpar.sys(M), rpar.sys(R)* und *setall.bat.*

2. Die Ereignisprotokolle ANWENDUNG und SYSTEM aus der Ereignisanzeige. Die Protokolle können über:

   (Windows2000/WindowsXP)
   Start–Einstellungen–Systemsteuerung–Verwaltung–Ereignisanzeige– *Anwendung* **/** *System*

   (WindowsNT4)
   Start– Programme – Verwaltung(Allgemein) – Ereignisanzeige-*Anwendung***/** *System*
   angezeigt werden und über den Menupunkt:

   (Windows2000/WindowsXP)
   Vorgang – Protokolldatei speichern unter ... / Aktion - Protokolldatei speichern unter ...

   (WindowsNT4)
   Protokoll-Speichern unter ...
   in eine Datei gespeichert werden.

3. Windows Diagnose-Bericht (komplett), aktivieren über:

   (WindowsNT4)
   Start–Programme–Verwaltung(Allgemein)–Windows NT-Diagnose
   und dann über den Menupunkt:
   Datei–Bericht speichern ... das Diagnoseprotokoll in eine Datei speichern.

   (Windows2000)
   Start–Einstellungen–Systemsteuerung–Verwaltung–Computerverwaltung,
   Unterverzeichnis *Systeminformationen* anklicken und über die MenuPunkte:
   Extras–Windows - Windows-Berichtsprogramm auswählen.
   Nach Klick auf **Systemdateiauswahl ändern** werden die Systemdateien zusammengestellt und zur Auswahl vorgelegt.
   Komplettauswahl über:
   Button **Alles markieren,** nach **OK** und **Weiter** erscheint der übliche SpeichernDialog.

   (WindowsXP)
   Start– Programme–Zubehör – Systemprogramme – Systeminformationen,
   Unterverzeichnis *Systemübersicht* auswählen und über die MenuPunkte:
   Datei–Speichern die SystemInfo-Datei abspeichern.

© IPKS GmbH 2007

10 Index

3941 27
3942 27
3945 28
3946 28
3947 28
3948 28
3949 28
3974MT 26, 27
3974MTNET 26, 27

ALE1 28
ALE1ONLY 86
ALEM 12, 133

BDE-Terminal 92
Beenden der Emulation 29
Belegungsmodus der Platten 37

Cache 19
CP1400 14, 27, 55
CPU-Typ 24, 30, 66
CS275 14, 27, 55

DEnet75 14
DF32 11, 14
DF42 11, 13, 14
DFCONS 138
Diskettenlaufwerk 38, 73
Dongle 15, 138
DR202 26, 28, 44, 76
DRCOM 25, 26, 28, 45, 77
DRSPOT 26, 41
DRSPOTS 26, 41
DRSPOTX 26, 28, 42, 78
DRUA 26, 28
Druckausgabe
  über Windows Druckertreiber 40
Druckausgabe in ein Windows Fenster 46, 80
Druckausgabe in eine Windows Datei 76
Druckausgabe in Windows Datei 41
Druckausgabe über Windows Druckertreiber 75
Drucker 9, 19, 138
Druckerbetrieb (parallel) 40, 75
Druckerbetrieb (seriell) 44, 45, 76, 77
Druckersteuerzeichen 138
DS074 27, 33
DS074NET 27, 35
DS075 27, 32
DS075 DISIT 26, 27
DSnet75 14
DU02 48
DU03 49
DU04 11, 27, 50, 51
DU05 11, 27, 51, 52, 54
DU05 Geräteemulation 96
DU06 11, 27, 53
DU06 Geräteemulation 98
DUST 3962 13
  Lichtwellenleiter 82, 85
DUST3964R 13, 28, 82
DUST3965 28
DUST3965R 13, 83
DUST3965R Geräteemulation 106
DUST3966 14, 28, 84, 123
DUST3966 Geräteemulation 109

EAES 27, 28, 65
EAMX 26, 64
EAMX1 91
Emulatorkern-Prozess 7
Emulator-Start 29
ETC M 14

FACO30 138
FestPlatte 9
FLOP 26, 28, 38, 73, 138
Floppy-Disk 10, 19, 138
FP0xx 38
FP0XX 26
FTNT 59, 88
FTNTD 61, 88
FTNTDX 61, 88
FTNTX 88

Gerätedefinition 25
Geräteemulation DS074 93
Gleitpunktarithmetik 8
Gleitpunktprozessor deaktivieren 67
Grafikkarte 19

Hard-Disk 19
Hauptspeicherausbau 24, 30, 66
Hauptspeicherrett-Datei 24, 30, 66

Installation 20

Kommunikation
  KS100, CS275, CP1400 55
Kommunikationsprozessor
  UCP-2 56
Konsolen-Emulation 9
KS N16
  (Anbindung TELEPERM) 92
KS100 27, 55

Leistungsumfang 8
Lieferumfang M 18
Lieferumfang M2000 16
Lieferumfang R 18

Magnetband-Kassetten-Laufwerk 10, 39, 74
Magnetwechselplatte WP256 74
MCSAVE 126
MEC-28R 67
MK82 26, 28, 39, 74
MODE
  (Belegmode Platten) 37
  (Belegmodus Platten) 72

ODBC 27
OP11 74
OP31
  MO Disk 26
OP31 Emulation 118
Optische Platte OP11 74
Optische Platte OP31 39

 © IPKS GmbH 2007

PARIF 12
PCCOPY 125
PE F7 12, 59
PE3600 12, 112
Performance 6
Programm WAIT4.EXE 127
PROMEA MX 13
PROMEA1 13
Promea-Zeitgeber 58, 86
Prozeßelement PE F7 59
Prozessor 19
PS048 26
PS048, PS049 37
PS049 26
Pseudogerät 62, 87, 89

Rechnerkopplung 13
Rechnerkopplung DU02 48
Rechnerkopplung DU03 10, 49
Rechnerkopplung DU04 50, 51
Rechnerkopplung DU05 51, 52, 54
Rechnerkopplung DU06 50, 53
Rechnerkopplung DUST3962 82
Rechnerkopplung DUST3964R über TCP/IP 82
Rechnerkopplung DUST3965R 83
Rechnerkopplung DUST3966 84

Schnittstellenkonzentrator 116
SCU-Service
  SwitchContrlUnit 124
Serielle Schnittstellen 19
Sichtgeräte 8
Sichtgeräte seriell angeschlossen 32, 68
Sichtgeräteemulation 8
Sichtgerätetypen 35, 70
SicView 129
SOLDIS
  MO Disk Copy 128
SPOOL75 14
Streamer 19

TE2000 AX über Netzwerk 70
TE2000 DX über Netzwerk 34, 35, 69, 70
Terminal 19
Terminal-Emulation lokal 33, 69
TrcView 139

UCP-2 56
Urladegerät 24, 31, 67

VDU2000 26
VICOM 121
virtuelle Konsole 24, 31

WINCC 62
Windows
  Win2000/WinXP 19
Windows-Zeit 58

ZE-Emulation 18
Zeitgeber 10, 121, 123
Zeitgeber 3691A 86
Zeitverhalten 121, 123
ZIG 58
ZIG1 26, 28